In compliance with the
Canadian Privacy Legislation
some supporting forms
may have been removed from
this dissertation.

While these forms may be included
in the document page count,
their removal does not represent
any loss of content from the dissertation.

University of Alberta

Extending Proportional Delay Differentiation to Cellular Wireless Networks

by

Kun Bai ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**.

Department of Computing Science

Edmonton, Alberta
Fall 2003

National Library of Canada
Acquisitions and Bibliographic Services
395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada
Acquisisitons et services bibliographiques
395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-87715-9
Our file Notre référence
ISBN: 0-612-87715-9

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

Canada

University of Alberta

Library Release Form

Name of Author: Kun Bai

Title of Thesis: Extending Proportional Delay Differentiation to Cellular Wireless Networks

Degree: Master of Science

Year this Degree Granted: 2003

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

Date: Sept. 26. 2003

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Extending Proportional Delay Differentiation to Cellular Wireless Networks** submitted by Kun Bai in partial fulfillment of the requirements for the degree of **Master of Science**.

Date: Sept. 26, 2003

Abstract

The Differentiated Services (DiffServ) architecture has recently received a growing interest as one of the main architectures adopted by the IETF for provisioning quality of service (QoS) for the future Internet. Since its conception, a number of forwarding per-hop behaviour schemes for providing various types of service differentiation have been proposed in the literature. Extending such Internet based QoS architectures to mobile users of the third and the fourth generations (3G and 4G) of wireless cellular networks becomes an enabling tool for mobile users to enjoy future Internet applications.

In this thesis we make a step toward achieving the above wireless-Internet integration by investigating the design of call admission control and packet scheduling algorithms to support the *proportional delay differentiation* per-hop behaviour for the uplink traffic (i.e., traffic from mobile users to a base station) in a cellular system employing the wide-band code-division multiple access (W-CDMA) air interface working in the frequency-division duplexing (FDD) mode.

The thesis contributions include a presentation of some recent work on providing network level QoS measures for the uplink traffic in 3G discrete sequence CDMA environments. Additionally, the thesis proposes and investigates the performance of three scheduling and admission control algorithms to achieve proportional delay differentiation. The performance measures of interest include the achieved proportional delay ratios, proportional delays, and throughput. Using simulation, we show improvements achieved by two of the proposed methods over a basic method with respect to achieving better approximation of the required proportional delay ratios. In certain cases, however, such improvements come at the expense of throughput degradation.

Acknowledgements

It is with deep gratitude and appreciation that I acknowledge the professional guidance and friendship of My supervisor Dr. Ehab S. Elmallah. His constant encouragement and support helped me to achieve my goal. My gratitude goes to the other members of the Network Group, Dr. Mike MacGregor and Dr. Janelle Harms. Their help and friendly guidance made two years a great learning experience.

I am grateful to the graduate faculty of the Department of Computing Science of the University of Alberta . Gratitude is also expressed to the members of my reading and examination committee, Dr. Mike MacGregor, Dr. Chintha Tellambura and Dr. Herb Yang.

I would also like to thank my beloved wife Ying Liu and some of the my friends, Jinhui Shen, Guang Li, Qiang Ye, Weiguang Shi, and Yuan Sha, for their constant help and encouragement.

Contents

1 Introduction 1

1.1 The Differentiated Services (DiffServ) Architecture 2

1.2 Forwarding Per-hop Behaviours 5

1.3 Overview of Wireless Personal Technologies 6

1.4 Thesis Contributions and organization 7

2 Background on the Cellular Wireless Environment 9

2.1 Overview of the UMTS System 10

2.2 The Large Scale Path Loss Model 11

2.3 Uplink Power Control 12

2.4 Distributed Asynchronous Power Control 15

2.5 Admitting Heterogeneous Traffic 17

2.6 Remarks on the Capacity of CDMA Cellular Systems 18

3 Related Work on QoS for Wireless CDMA Cellular Networks 21

3.1 Work on Virtual Bandwidth 21

3.1.1 Notations, Definitions, and Basic Observations 23

3.1.2 Details of the Admission Control Algorithm 28

3.1.3 Overview of the Obtained Results 31

3.1.4 Discussion 32

3.2 Work on Weighted Fair Scheduling 32

3.2.1 Background on GPS 33

3.2.2 Basic Assumptions and Observations 34
3.2.3 The CDGPS Algorithm 36
3.2.4 The A-CDGPS Algorithm 37
3.2.5 Overview of the Results 38
3.2.6 Discussion 39

4 Simulation Environment **40**
4.1 Overview of the Communication Protocol 40
4.2 Simulation Parameters 41
4.2.1 User Mobility Parameters 41
4.2.2 CDMA paramters 42
4.2.3 Traffic Parameters 43
4.2.4 Setting the Maximum Number of Mobiles 44

5 A Basic Proportional Delay Differentiation Scheduler **46**
5.1 Background 46
5.1.1 A Basic PDD Scheduler 49
5.2 Simulation Results 50
5.2.1 Average Class Delay 51
5.2.2 Average Class Delay Ratio 51
5.2.3 Average Class Throughput 52
5.3 A Basic Call Admission Control Scheme 53
5.4 Simulation Results using the Basic CAC Algorithm 54
5.4.1 Average Delay 54
5.4.2 Throughput 56
5.5 Conclusions 56

6 An Improved CAC Algorithm **57**
6.1 Improved Call Admission Control Scheme 57
6.2 Performance Comparison of Two CAC Schemes 60

6.2.1 Throughput 60
6.3 Conclusions 61

7 Performance Enhancement Using Adaptive Parameter Setting **62**
7.1 Time-Dependent Priority Queues (TDPQ) 63
7.2 Using the TDPQ System 64
7.3 Solving For the Required Delay Ratios 66
7.4 Case Studies 68
7.5 Results on Average Delay Ratio, Average Delay and Throughput 70
7.5.1 Average Delay Ratios 70
7.5.2 Average Delay of Classes 71
7.5.3 Throughput of Classes 72
7.6 Conclusions 73

8 Conclusions and Future Works **74**
8.1 Summary 74
8.2 Future Works 75

List of Tables

4.1 Mobility Parameters 42
4.2 CDMA Parameters 43
4.3 Traffic Parameters 44
4.4 Capacity for different spreading factor 45

List of Figures

1.1 DiffServ Architecture 3

2.1 UMTS Architecture 10

2.2 Interference of the uplink in a multi-cell environment 13

4.1 Control Slot Model 40

4.2 Multi-cell environment 42

5.1 Queueing model for Proportional Delay Differentiation scheduling 48

5.2 Average Class Delay Without CAC 51

5.3 Average Class Delay Ratio Without CAC 52

5.4 Average Class Throughput Without CAC 53

5.5 Average Delay without the CAC Algorithm 55

5.6 Average Delay with the Basic CAC Algorithm 55

5.7 Throughput without the CAC 56

5.8 Throughput with the Basic CAC 56

6.1 Average Delay of Algorithms 1 and 2 (same as Figure 5.6) 60

6.2 Average Delay of Algorithms 1 and 3 60

6.3 Throughput with the Basic CAC Algorithm 61

6.4 Throughput with the Improved CAC Algorithm 61

7.1 Results using the Δ_i values. 69

7.2 Results using the b_i values 69

7.3 Results using the Δ_i values. 69
7.4 Results using the b_i values 69
7.5 Average Delay Ratio under Different System Load 71
7.6 Average Delay under Different System Load 72
7.7 Class Throughput – Δ_i 72
7.8 Class Throughput – b_i 72

List of Abbreviation

3G	Third Generation Wireless Cellular System
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BS	Base Station
CDMA	Code Division Multiple Access
CAC	Call Admission Control
DDP	Delay Differentiation Parameter
DiffServ	The IETF Differentiated Services Architecture
FDMA	Frequency Division Multiple Access
GSM	Global System for Mobile communications
IETF	The Internet Engineering Task Force
IntServ	The IETF Integrated Services Architecture
IMT-2000	The International Mobile Telecommunications 2000
ITU	The International Telecommunication Union
MAI	Multiple Access Interference
PDD	Proportional Delay Differentiation
QoS	Quality of Service
SIR	Signal to Interference Ratio
SMS	Short Message Service
TDMA	Time Division Multiple Access
TDP	Time Dependent Priority
UMTS	Universal Mobile Telecommunications System
WCDMA	Wideband CDMA
VBR	Variable Bit Rate

List of Symbols

(g, G) – –Path Gain between Mobile and BS 13

P_i – – Received power from mobile i 13

W – – Chip rate in WCDMA 13

$\frac{E_b}{I_o}$ – – bit-energy-to-interference-power-ratio 15

(r, R) – –Transmission rate in uplink 15

C_i – – Share of soft system capacity for mobile i 18

$f(t)$ – –Ratio of intracell and intercell interference 18

Δ_i – – Proportional delay differentiation control parameter for class i 46

C_{res} – –Residual system capacity 58

ρ_i – – Share of system utilization for class i 63

λ_i – – Average arrival rate for class i 63

W_i – – Average waiting-time for class i 64

W_0 – –Residual service time 64

$\overline{x^2}$ – – Second moment of service time 64

$\overline{x}$ – – Average service time 64

b_i – – Adaptive delay control parameter for class i 64

Chapter 1

Introduction

The use of mobile personal wireless communication devices to obtain any time, any where voice and data services is increasing steadily world wide. For the Internet access, it is expected that in the near future the number of mobile terminals connected to the Internet by wireless means will outnumber the current conventional wired office terminals. The wireless end user access will be enabled by means of wireless local area networks (WLANs), as well as wireless cellular networks. WLAN technologies have the advantage of being relatively cheap, while cellular wireless networks have the advantage of offering wider geographical area coverage, advanced voice/data services and accounting offered by specialized phone and network providers, and (when technologies achieve world wide convergence) the possibility of global roaming.

Enjoying future Internet applications, however, requires the extension of future Internet quality of service (QoS) architectures to wireless domains. This thesis considers such wireless-Internet QoS integration by investigating methods for supporting one type of the IETF Differentiated Services (DiffServ) architecture to third generation (3G), and beyond, cellular networks.

To explore and motivate this research direction further we organize this chapter as follows. Section 1.1 briefly describes the Differentiated Services (DiffServ) architecture. Section 1.2 gives us an overview of some well known forwarding per-hop behaviours. Section 1.3 brings us a very brief introduction of the progress of the wireless personal technologies. Section 1.4 highlights the organization and contributions of the thesis.

1.1 The Differentiated Services (DiffServ) Architecture

We are familiar that the Internet is constructed using a simple protocol, the *Internet Protocol* (IP), which mostly provides a best-effort service. Due to the lack of guaranteed quality of service in the Internet, the IETF workgroups have been actively seeking suitable architectures to support QoS. The most current mechanisms of QoS are the *Integrated Services* (IntServ) architecture and the *Differentiated Services* (DiffServ) architecture [4].

To provide differentiated services, the simplest way is that the differentiation can be achieved based on appropriate pricing (higher classes are more expensive), or based on the capacity provisioning (higher classes get more capacity). However, such approaches can not always provide consistent differentiation between classes because the offered service depends on the system load. In general, any useful approach should be:

- Controllable: meaning that the network operators should be able to adjust the relative QoS difference between classes based on their criteria.
- Predictable: meaning that the differentiation should be consistent without being dependent on the system load variations.

In the following, we give an overview of the DiffServ architecture. DiffServ [5] has been developed to alleviate the scalability problems of the Integrated Services (IntServ) model and, at the same time, to support *quality of service* (QoS) capabilities.

In IntServ, we focus on individual flows. RSVP [35] provides per-flow reservation and each router keeps per-flow state. In contrast, DiffServ classifies traffic into a small number of classes, and allocates resources for each class. Packets are forwarded based on the class information encoded in the packet header. There is no need to reserve resources for individual flows or to keep per-flow state in each router. As a result, DiffServ does not suffer from the IntServ scalability problems. By allocating different resources to each class, DiffServ provides each class with different levels of service. In DiffServ, there is a contract between the customer and the service provider, which defines what kind of service the customer wants to receive. This contract is called the *service level agreement* (SLA).

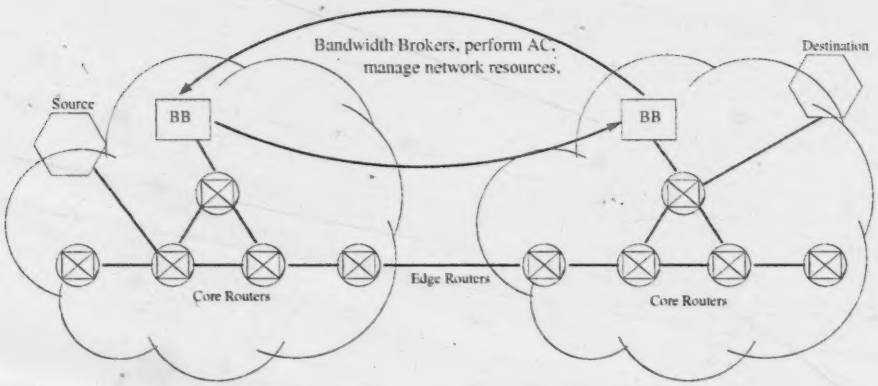

Figure 1.1: DiffServ Architecture

A differentiated-services domain (DS domain) normally consists of one or more networks under the same administration, such as an organization's intranet or an ISP. There are two types of routers in each domain: *edge* routers and *core* routers. The edge routers include *egress-edge* router, and *ingress-edge* router (which interconnect a DS domain to other domains). The core routers only connect to other routers and the *bandwidth broker* (BB) in the same domain. In DiffServ, traffic policing is done at the edge and the class-based forwarding is done in the core.

Figure 1.1 shows part of the DiffServ architecture. DiffServ edge routers perform three functions: classication, marking, and conditioning. The classifier selects packets and maps each packet to a particular forwarding behaviour. The selected packets are marked with a particular differentiated service codepoint (DSCP) [19]. Each codepoint is a 6-bit number encoded in the packet header, and it is used by the core routers to decide the forwarding treatment for the packet.

The conditioner in the edge routers measures the traffic stream against a traffic profile, which comes from the SLA. Depending on whether the traffic stream meets the SLA specifications or not, the packets are classified as either in-profile or out-of-profile. Different actions such as dropping, shaping, or remarking may be performed on out-of-profile packets.

The core routers perform fewer operations than the edge routers; they apply the for-

warding behavior by mapping the DS codepoint marked by the edge routers to one of the implemented per-hop behaviours (PHBs). In each DiffServ domain, there is another component called a bandwidth broker [20]. BBs have two responsibilities: resource management for each domain, and message passing to the adjacent domain's BB. If bandwidth allocation is desired for a flow, a request will be sent to the BB. The BB will then verify that there exist sufficient unallocated resources to meet the request. If so, the BB informs the leaf router with the flow's information to deliver the service to the flow at the time the service is needed. If the destination is outside the requester's domain, the requester domain's BB informs an adjacent domain's BB that it will use this amount of resource allocation.

Based on the above discussion, we see that the essential characteristics of DiffServ include:

- **Handling of aggregated traffic instead of individual flow.** Resources are allocated to each class. For individual flow, there is no absolute guarantee.
- **Class-based forwarding in the core.** In the core routers, there is no per-flow state kept, as there is in the IntServ. Core routers forward the packets, by mapping the codepoint encoded in the packet header, to one of the implemented PHBs. The selected PHB decides which kind of forwarding treatment the packet can receive.
- **Traffic policing at the edge.** The edge routers perform classication, marking, and conditioning. This ensures that all the packets coming into the core network are marked properly for class-based forwarding.
- **Resource provisioning rather than reservation.** In the IntServ, the RSVP protocol reserves resources along the path for each flow if the resource is available. However, in DiffServ, resources are allocated to each class, and all the packets belonging to the same class compete for the total resource allocated to the class.

1.2 Forwarding Per-hop Behaviours

Many DiffServ per-hop forwarding behaviours have been studied. Although they all conform to the same DiffServ architecture of figure 1.1, they differ in the specific services provided, and in their implementation mechanisms. We outline some well known per-hop forwarding behaviours including *expedited forwarding* (EF), *assured forwarding* (AF), and the proportional delay differentiation.

1. **Expedited Forwarding [2] :** Expedited Forwarding (EF), which is also called *premium service*, is defined as a forwarding treatment for a traffic aggregate, where the departure rate of the aggregate's packets from any DS node must equal (or exceed) a configurable rate. Moreover, the EF traffic should always receive the configurable rate independent of the intensity of other traffic types. EF service provides the equivalent of a dedicated link with a fixed bandwidth. A lower-bound of service rate is guaranteed for EF packets. As a result, EF service could provide effective support for real-time applications.

2. **Assured Forwarding [1]:** Assured Forwarding PHB is suggested for applications that require better reliability than the best-effort service. There are four classes of service, and within each class, there are three drop precedences. There are also a number of factors affecting the performance of Assured Forwarding including bandwidth management, buffer management, and fairness between different traffic types. Considerable research has been done on such aspects.

3. **Proportional Differentiation [8] :** The proportional differentiation model makes certain class performance metrics proportional to differentiation parameters that the network operator chooses. If, say, q_i is such a performance measure for class i, the proportional differentiation model imposes constraints of the following form for all pairs of classes:

$$\frac{q_i}{q_j} = \frac{c_i}{c_j} \; (i, j = 1...N),$$

where $c_1 < c_2 < ... < c_N$ are the generic quality differentiation parameters. So, even though the achieved absolute value of q_i for class i varies with the traffic load, the ratios between the q_i values for different classes remain fixed and controlled by the network operator, independent of the class loads.

Proportional delay differentiation (see, for example, [8, 16]), considered as useful proportional differentiation model, is an important PHB. An ideal PDD scheduler should satisfy the following relation: if $\overline{d_i}(t)$ denotes the average delay incurred by class i packets over a time window of length t during which the classes are continuously backlogged, then for any two classes i and j

$$\frac{\overline{d_i}(t)}{\overline{d_j}(t)} = \frac{\Delta_i}{\Delta_j}$$

where Δ_i is a positive weight specified for each class.

1.3 Overview of Wireless Personal Technologies

The recent technological advances of wireless personal systems and networks are explained in many recent books (see for example, [10], [11], [22], [23], or [29]). Below we summarize some aspects of such advances which show the progress from providing voice-centric services in the past, to voice and low bit data rates at present, to voice and higher data rates in the near future. Among many other factors, the advances also reflect an increase in the traffic carrying capacity of the systems.

- The *first generation* (1G) wireless network systems appeared in the early to mid 1980's, and used analog technology to provide voice only services over a limited geographical area.

- The second generation (*2G*) wireless systems use digital technology to provide better voice quality, higher system capacity, global roaming capability as well as lower power consumption. Examples of 2G systems include GSM (based on TDMA technology) in Europe, and IS-95 (based on CDMA technology) in United State. Im-

proved 2G systems provide support for simple data services like *short message service* (SMS).

- In order to guarantee a smooth transition from 2G towards the *third generation* (3G), the IMT-2000 was set up by the *International Telecommunication Union* (ITU) to harmonize the different proposed 3G standards. Of interest to us are systems that satisfy the IMT-2000 requirements which include CDMA2000 (North American) and W-CDMA (Europe and Japan). Such systems are designed to provide high transmission data rates and multimedia communication that include up to 2 Mbps data for stationary users, 384 Kbps for low-speed moving users, and 144 Kbps for high-speed moving users. The meterial included in this thesis applies to both standards.

1.4 Thesis Contributions and organization

The main focus of the thesis is on developing and investigating the performance of call admission control and scheduling strategies to extend the PDD model to the uplink traffic in next generation wireless systems. In this thesis, we only consider the transmission on the uplink (i.e., from mobile users to base stations). We use simulation as the main tool in our performance study. Toward achieving the above goal, the rest of the thesis is organized as follows.

Chapter 2 reviews some fundamental and technological aspects relevant to a 3G CDMA wireless cellular environment. The information presented in chapter 2 form the basis of the designs proposed in the thesis.

Chapter 3 discusses previous work on providing QoS over wireless CDMA networks. This includes work on virtual bandwidth (VB), and fair packet scheduling in CDMA soft-capacity environments.

Chapter 4 describes our simulation environment and the parameters used in our study.

Chapter 5 develops a scheduling algorithm and a call admission control algorithm to prodive PDD.

Chapter 6 develops an improved CAC algorithm intended to maximize throughput, and also improve the fairness among priority classes. Simulation results are also presented.

Chapter 7 seeks to achieve tighter control on delay ratios by focusing on the dynamics of the priority queueing taking place at the users side. In particular, we try to relate the relative weights used by the scheduling algorithm (to induce service differentiation) with the resulting relative delays.

The main contributions of this thesis are:

- a detailed presentation of some selected recent work on providing QoS for uplink traffic in 3G CDMA networks. The selected results include the definition and use of the concept of Virtual Bandwidth (VB) , and an application of the generalized processor sharing service discipline,
- proposing and investigating three scheduling and call admission control algorithms to achieve proportional delay differentiation in CDMA-based cellular environments.

Chapter 2

Background on the Cellular Wireless Environment

In the previous chapter we motivated the research direction aiming at extending the proportional delay differentiation per-hop behaviour of the DiffServ architecture to mobile users served by a 3G (or 4G) wireless cellular systems. In this chapter we review some important fundamental and technological aspects relevant to a CDMA wireless cellular environment. The information presented here form the basis of the designs proposed in the thesis.

The chapter is organized as follows. Section 2.1 gives a brief overview of the architecture of the UMTS/UTRAN system that makes both of the circuit-switched services and the packet-switched services available to the mobile end user. Section 2.2 discusses the log-normal shadow fading model for large scale path loss that is used in our modeling and simulation setup. Section 2.3 discusses a necessary condition for the traffic from a mobile user to be acceptable at the base station, and the role of power control in achieving such necessary condition. Section 2.4 digresses and discusses distributed asynchronous power control. Based on the information in Section 2.3, Section 2.5 derives an important necessary condition for a set of heterogeneous traffic connections to be acceptable at the base station. In Section 2.6 we conclude with some remarks on the capacity of CDMA cellular systems.

2.1 Overview of the UMTS System

A UMTS/UTRAN network consists of three interacting domains: *Core Network* (CN), *UMTS Terrestrial Radio Access Network* (UTRAN) and *User Equipment* (UE). The main function of the core network is to provide switching, routing and transit for user traffic. Core network also contains the databases and network management functions.

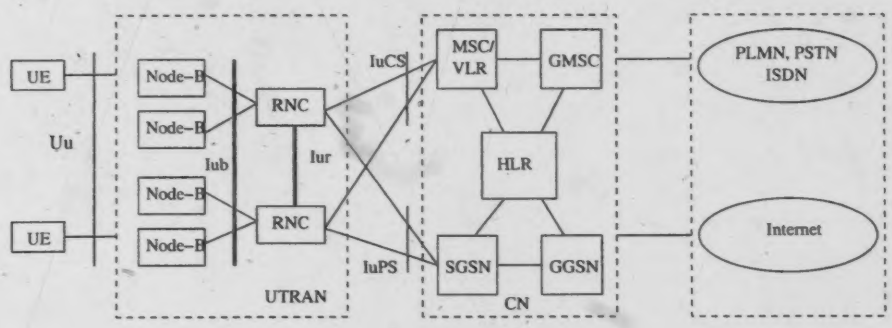

Figure 2.1: UMTS Architecture

The basic Core Network architecture for UMTS is based on GSM/GPRS network architecture. All equipment has to be modified for UMTS operation and services. The UTRAN provides the air interface access method for UE. A base station is referred as Node-B and the control equipment for Node-B's is called *Radio Network Controller* (RNC). Figure 2.1 illustrates the architecture of a UMTS system.

The Core Network is divided into circuit switched and packet switched domains. Circuit switched elements are: the *Mobile services Switching Centre* (MSC), the *Visitor Location Register* (VLR), and the Gateway MSC. The packet switched elements are: the *Serving GPRS Support Node* (SGSN), and the *Gateway GPRS Support Node* (GGSN). Some network elements, like *Equipment Identity Register* (EIR), *Home Location Register* (HLR) and *Authentication Center* (AUC), are shared by both domains.

The *Asynchronous Transfer Mode* (ATM) is defined for UMTS core transmission. The *ATM Adaptation Layer type 2* (AAL2) handles circuit switched connection and packet connection protocol AAL5 is designed for data delivery.

The architecture of the Core Network may change when new services and features are

introduced. *Number Portability DataBase* (NPDB) will be used to enable user to change the network while keeping their old phone number. *Gateway Location Register* (GLR) may be used to optimise the subscriber handling between network boundaries. MSC, VLR and SGSN can merge to become a UMTS MSC.

Wide band CDMA technology has been selected for the UTRAN air interface. UMTS WCDMA is a Direct Sequence CDMA system where user data is multiplied with quasi-random bits derived from spreading codes. For example, TDD WCDMA uses spreading factors $4 - 512$ to spread the base band data over 5 MHz band. In UMTS, in addition to channelisation, codes are used for synchronisation and scrambling. WCDMA has two basic modes of operation: *Frequency Division Duplex* (FDD) and *Time Division Duplex* (TDD). For FDD mode, UMTS frequencies is assigned as following: $1920 - 1980$ MHz (uplink) and $2110 - 2170$ MHz (downlink).

UMTS offers teleservices and bearer services, which provide the capability for information transfer between access points. It is possible to negotiate and renegotiate the characteristics of a bearer service at session or connection establishment and during ongoing session or connection. Both connection oriented and connectionless services are offered for point-to-point and point-to-multipoint communication. Bearer services have different QoS parameters for maximum transfer delay, delay variation and bit error rate (BER). Offered data rate targets are: 144 Kbps rural outdoor, 384 Kbps urban outdoor, and 2048 Kbps indoor and low range outdoor.

2.2 The Large Scale Path Loss Model

To model path loss, the free space model is useful when the transmitter and receiver are in line of sight. In urban environments, buildings, trees, cars, etc. obstruct the waves and cause additional attenuation than in free space. The radio waves undergo reflection, diffraction and scattering. Due to the complexity of the environment it is difficult to derive theoretically a model which predicts the net resulting wave as a function of distance in a real environment.

The theoretical and practical findings in [22] indicate that the average received signal strength decreases logarithmically with distance and path loss can be described by the following log-normal shadowing model:

$$PL(d)\,[dB] = \overline{PL}(d) + \chi_\sigma = \overline{PL}(d_0) + 10n\log(\frac{d}{d_0}) + \chi_\sigma \qquad (2.1)$$

where d_0 is a reference point, $\overline{PL}(d_0)$ is the average path loss (in dB) at d_0, n is an exponent varying with different environments, and χ_σ is a 0 mean Gaussian distributed random variable with standard deviation in dB. In our simulation work, we set $\sigma = 4$ dB, $n = 3.7$, $d_0 = 1$ m, and $\overline{PL}(d_0) = 128.1$ dB. Therefore, the above equation simplifies to:

$$PL(d)\,[dB] = 128.1 + 37.6\log(d) + \chi_\sigma \qquad (2.2)$$

2.3 Uplink Power Control

In the paired FDD mode a cluster of neighbouring cells can use one 4.096 MHz band (in the range 1920-1980 MHz) for all uplink communications in all cells, and a different 4.096 MHz band (in the range 2110-2170 MHz) for all downlink communications in all cells. Figure 2.2 illustrates the signals received by the base station in the central cell. In particular, the figure illustrates that the received signal of any particular mobile user must overcome an interference power made of the signals from all other mobiles in the same cell, as well as all mobiles from other cells.

Separation of users in different cells is achieved by using cell-specific scrambling codes, while separation among users in the same cell is done using user-specific spreading codes (see, for example, [7]).

A basic inequality that gives a necessary condition for the received power from mobile user i to achieve a certain BER (say, BER= 10^{-6}) (see, for example, [12]) is

$$(\frac{E_b}{I_0})_i \leq \frac{W}{R_i}\ \frac{P_i}{P_\Sigma^i + I_{ext} + n_0} \qquad (2.3)$$

where

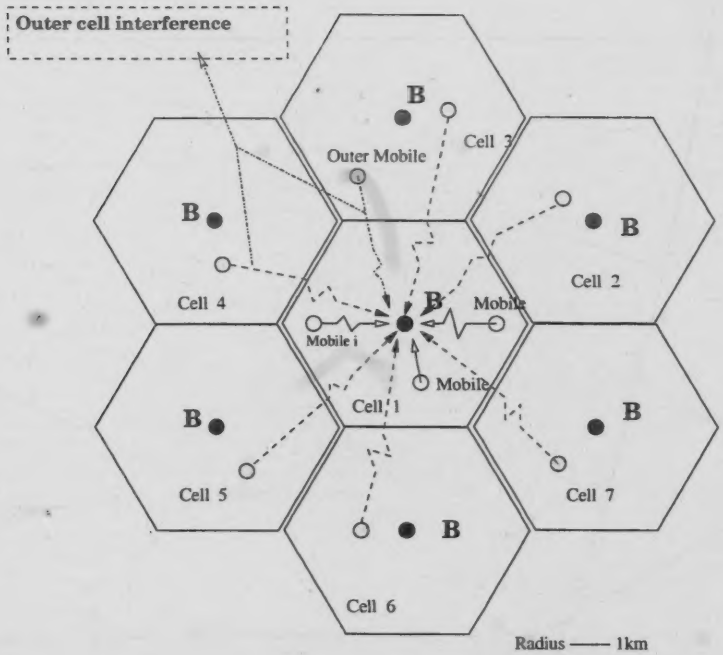

Figure 2.2: Interference of the uplink in a multi-cell environment

- $(\frac{E_b}{I_0})_i$ is a target bit-energy to interference ratio required to achieve a desired BER for mobile i at the base station (e.g. for a given receiver structure at the base station, we may need $\frac{E_b}{I_0} = 7$ dB to achieve BER= 10^{-6}).

- W is the chip rate of the discrete-sequence CDMA system ($W = 4.096$ Mcps in the W-CDMA standard).

- R_i is the mobile host transmission rate of the encoded data.

- W/R_i is called the *spreading gain* (the transmitter of the mobile host spreads each data bit into W/R_i chips).

- $\frac{P_i}{P_\Sigma^i + I_{ext} + n_0}$ is the signal-to-interference (SIR) ratio for mobile i signal.

- P_i is the received power from mobile i at the serving base station. That is, if g_i denotes the radio channel gain (a random variable), and P_i^T denotes the mobile transmission power then $P_i = g_i P_i^T$.

- $P_{\Sigma}^{i} = \sum_{j \neq i} P_j$ is the sum of all received power from all other mobile hosts in the target cell
- I_{ext} is the sum of all received power from all mobiles outside the target cell
- n_0 is the thermal noise power at the base station receiver.

We now outline more aspects of the above inequality through the following remarks.

1. The term power control refers to adjusting each mobile's transmission power in a time varying wireless channel so as to achieve an acceptable SIR at the base station while using the minimum possible transmission power. This has the effect of
   - minimizing the interference power caused by a mobile's transmission which affects other mobiles; this leads to increasing the capacity of the cell, and
   - conserving the mobile's battery power.

   Under perfect power control, the above inequality holds as an equality.

2. Additionally, in the UTRA environment, power control also aims at maintaining equal received power at a base station from all mobiles under its control. So, under perfect power control, if N is the number of served mobiles then the above inequality simplifies to
$$(\frac{E_b}{I_0})_i = \frac{W}{R_i} \frac{P_i}{(N-1)P_i + I_{ext} + n_0}.$$

3. (a numerical example) Suppose, for example, that outside cell interference and noise $I_{ext} = 0.3 \cdot P_{\Sigma}^{i}$, $W = 4.096$ Mcps, $(\frac{E_b}{I_0}) = 7$ dB. When mobile transmission rate is assigned at 128 kbps, the system can support up to 6 mobiles. As we can see from above equation, increasing the transmission rate of the encoded data to the i_{th} user results in decreasing the spreading gain (W/R).

4. We note that the minimum transmission power can be computed by solving a linear system of equations.

5. Although centralized power control is used in advanced CDMA systems, such as UMTS, there has been numerous research work on achieving power control through the use of distributed algorithms (where each mobile independently executes a simple algorithm). Such environments require less expensive hardware. Although the results of this thesis do not apply directly to such environments, it is an interesting topic to invesigate the performance of the desired algorithms in such environments. We will leave this aspect as a possible future research work.

2.4 Distributed Asynchronous Power Control

In [33], the author shows a unifying framework for situations in which a simple distributed asynchronous algorithm is guaranteed to converge to minimum power levels. In this section, we explain the result further by reproducing the following key points from [33]:

1. The system model includes N users, and M base stations that use a common radio channel. The transmitted power of user j is denoted p_j, and the gain of user j to base station k is denoted h_{kj}. Thus, at base station k, the received signal power is $h_{kj}p_j$ and the interference seen by the transmission of user j at base k is $\sum_{i \neq j} h_{ki}p_i + \sigma_k$, where σ_k denotes the receiver noise power at base k.

   Hence, under power vector $\mathbf{p} = (p_1, p_2, ..., p_N)$, the SIR of user j at base station k is $p_j \mu_{kj}(\mathbf{p})$ where

$$\mu_{kj}(\mathbf{p}) = \frac{h_{kj}}{\sum_{i \neq j} h_{ki}p_i + \sigma_k} \tag{2.4}$$

2. Yates [33] considered the interference constraints obtained by the following strategies of obtaining the signal of user j.

   (a) **Fixed Assignment**: where user j is assigned to base station k by outside means, for example, based on the received signal strength of the base station pilot tone signal,

(b) **Minimum Power Assignment**: where at each step of the iterative power control algorithm, user j is assigned to the base station at which the SIR is maximized

(c) **Macro Diversity**: where the received signal of user j at all base stations are combined using a maximal ratio combining method,

(d) **Limited Diversity**: where the received signal of user j is combined from a subset of cardinality d_j, $d_j \leq M$, of the base stations at which user j has highest SIR,

(e) **Multiple Connection Reception**: where user j is required to maintain an acceptable SIR in at least a certain number $d_j \leq M$ of base stations.

3. To discuss the result, we take the second strategy (denoted MPA) as an example. Here, the objective of user j is to set p_j so that $p_j \mu_{\ell j} \geq \gamma_j$ is realized at some base station ℓ. That is, user j wants to achieve $\max_k (p_j \mu_{kj}(\mathbf{p})) \geq \gamma_j$, which can be written as:

$$p_j \geq \min_k \frac{\gamma_j}{\mu_{kj}(\mathbf{p})} \tag{2.5}$$

Then the right-hand-side of above equation can be viewed as the interference function, denoted $I_j^{MPA}(\mathbf{p})$, from other users that user j would like to overcome (e.g., user j would like to achieve $p_j \geq I_j^{MPA}(\mathbf{p})$).

4. Assuming that an interference function $I_j(\mathbf{p})$ (like $I_j^{MPA}(\mathbf{p})$ above) satisfies the following properties:

- positivity: $I(\mathbf{p}) > 0$ (guaranteed by the nonzero background receiver noise),
- monotonicity: If $\mathbf{p} \geq \mathbf{p}'$, then $I(\mathbf{p}) \geq I(\mathbf{p}')$.
- scalability: For all $\alpha > 1$, $\alpha I(\mathbf{p}) > I(\alpha \mathbf{p})$. (That is, if user j has an acceptable connection under power vector $\mathbf{p}$, then user j will have an acceptable connection when all powers are scaled up uniformly)

then [33] has shown that if each mobile j executes the simple iteration $p_j(t+1) = I(\mathbf{p})$, then all mobiles will converge to using the minimum acceptable power levels if a feasible solution exists, and will detect infeasibility otherwise.

5. At last, it is nice to note that if $R_k(\mathbf{p}) = \sum_j h_{kj}p_j + \sigma_k$ denotes the total received power at base station k, then $\mu_{kj}(\mathbf{p})$ can be written as

$$\mu_{kj}(\mathbf{p}) = \frac{h_{kj}}{R_k(\mathbf{p}) - h_{kj}p_j}. \tag{2.6}$$

So, the power controlled systems mentioned above can be implemented by each user knowing only its own uplink gain and the total received power at base station. Thus, it is not necessary to know all uplink gains or transmitted powers of the other mobiles. This suggests that the algorithms may be suitable for distributed asynchronous implementation in real system in which users must perform updates with wrong or outdated interference measurements.

2.5 Admitting Heterogeneous Traffic

In this section we derive a necessary condition for a set of N (heterogeneous) traffic requests to qualify for admission at the base station. The condition is the basis of the admission control and scheduling algorithms devised in the thesis. The ith traffic request is characterized by its need for a transmission rate R_i, and the need to achieve at least a certain (E_b/I_0) lower limit denoted ξ_i. In the derivation, let

- P_i be the received power from user i
- $P_\Sigma = \sum_{i=1}^{N} P_i$
- I_{ext} the total external interference power from all users in all neighbouring cells plus thermal power noise at the base station receiver.

The necessary condition, given by relation (2.3), when applied to each of the N traffic flows, can be written as:

$$\xi_i \leq \frac{W}{R_i} \frac{P_i}{P_\Sigma - P_i + I_{ext}} \quad i = 1, 2, ..., N.$$

(Note: the above conditions are not sufficient conditions for the traffic from all of the N mobiles to be decoded correctly at the base station because some mobiles may not have enough transmission power to achieve a sufficiently large P_i at the base station.)
The above inequality can be written as

$$\frac{W}{\xi_i R_i} + 1 \geq \frac{P_\Sigma + I_{ext}}{P_i}.$$

Equivalently, $\frac{1}{\frac{W}{\xi_i R_i}+1} \leq \frac{P_i}{P_\Sigma + I_{ext}}$.

We view (the dimensionless quantity) $\frac{1}{\frac{W}{\xi_i R_i}+1}$ as the share of user i from the available bandwidth, and denote it as C_i. So, the above inequality can be written as

$$C_i \leq \frac{P_i}{P_\Sigma + I_{ext}}$$

Summing over all N admitted traffic requests, we get

$$\sum_{i=1}^{N} C_i \leq \frac{P_\Sigma}{P_\Sigma + I_{ext}}.$$

As a further simplification, if we assume that the ratio $f(t) = \frac{I_{ext}}{P_\Sigma}$ can be estimated at the base station at any time t, then the above equation takes the simpler form

$$\sum_{i=1}^{N} C_i \leq \frac{1}{1 + f(t)}. \tag{2.7}$$

The above relation is used in Chapters 5, 6, and 7 in the design of our resource management algorithms.

2.6 Remarks on the Capacity of CDMA Cellular Systems

During the years of late 1980s and early 1990s there has been a considerable debate about the call capacity (the number of voice calls that can be served concurrently) of the CDMA cellular systems in comparison to TDMA systems (such as the well known GSM system). In this section we mention a few remarks in this direction.

1. Consider a simple case of a single cell (with no outside interference) with N mobile users all requiring the same $\frac{E_b}{I_0}$ value. It is also assumed that perfect power control is applied so that all the uplink signals are received at the same power level. From relation (2.3) we get:

$$(\frac{E_b}{I_o}) = \frac{\frac{W}{R}P}{(N-1)P + I_{ext} + n_0}. \tag{2.8}$$

Let $\eta = I_{ext} + n_0$, then we get:

$$(\frac{E_b}{I_o}) = \frac{\frac{W}{R}}{(N-1) + \frac{\eta}{P}} \tag{2.9}$$

This implies that the capacity in terms of mobile users can be given by:

$$N = 1 + \frac{W/R}{E_b/I_o} - \frac{\eta}{P} \tag{2.10}$$

Equation (2.10) allows a simple comparison of the CDMA system with the other multiple -access systems. Let us have a simple example here. Consider a bandwidth of 4.096 MHz and a bit rate of 32 Kbps using voice coders. Let's assume that a minimum E_b/I_o of 5 (7dB) is required to acheive adequate performance (BER of 10^{-3}). We ignore η for the moment. The number of mobile user one cell can afford is:

$$N = 1 + \frac{4.096MHz/32kbps}{5} = 26 \tag{2.11}$$

For a GSM system in the same situation, the number of mobile users can be 126. However, the above equation does not take into account the effect of the on-off periods during a normal voice conversation which can be exploited using variable rate encoders.

2. *Sectoring* (or Sectorization) is one technique to increase the capacity of the CDMA cell. Here instead of using an omni directional antenna, the cell is divided into 3 or

6 sectors, each sector is served by a directional antenna. Assuming three directional antennas having 120^o effective beam widths are employed. Now, the interference sources seen by any of these antennas are approximately one-third of those seen by the omni-directional antenna. This reduces the interference term in the denominator of equation (2.9) by a factor of 3 and the number of users (N) is approximately increased by the same factor.

Chapter 3

Related Work on QoS for Wireless CDMA Cellular Networks

Since their beginning as candidate air interfaces for 3G system, the physical and the link level layers in the CDMA-based proposals (W-CDMA and CDMA2000) have been designed to give the designer flexibility in achieving various QoS measures for real time traffic. The amount of QoS literature on these two layers is voluminous and continues to grow at a fast pace. Our interest in this thesis is on supporting QoS at the level of packets and flows.

To gain an insight into some particular techniques that has been considered thus far to achieve QoS at the packet level we have identified two groups of results to present. The basic method used in the first group relies on the concept of *virtual bandwidth* (VB) and is introduced in Section 3.1. The basic methods used in the second group rely on applying the concept of the *generalized processing sharing* (GPS) service discipline to a CDMA environment; these methods are discussed in Section 3.2. Each section is followed by a brief discussion to position our work with respect to the described previous work.

3.1 Work on Virtual Bandwidth

In the Ph.D. work of [24], the author introduced the concept of virtual bandwidth for the uplink traffic in wireless cellular CDMA networks, and used the concept in designing an admission control (*AC*) scheme for heterogeneous traffic that takes certain QoS measures

into account.

In this section we present the details of a special case of the admission control scheme discussed in [24] and [25] (called the one-channel case). The material is based on chapters 1 to 3 (pages 1 to 60) in [24]. The basic ideas summarized here are the basis of the work in [25], [26], [27], and [28]. We follow (and in certain parts quote) [24] and [25] very closely, but organize the material to address the following questions: What QoS measures is being considered? How does the AC scheme work? and How does the AC achieve the desired QoS measures?

In the presentation, N denotes the number of active mobile hosts/users in the cell under consideration. Each user has one queue for the packets to be transmitted to a common base station. For each mobile host i, the following two QoS parameters serve as input to the admission control algorithm:

1. a user-defined constraint D_i on the average delay incurred by packets queued in the transmission queue, and

2. a user-defined constraint P_L^i on the wireless channel packet loss probability.

The general objective of the above formulation is to be able to serve a variety of heterogeneous traffic types (e.g., delay sensitive voice traffic, delay insensitive and loss intolerant file transfer traffic, delay sensitive and loss tolerant stored media traffic, etc.) by setting the QoS constraints D_i and P_L^i to suitable values. The admission control scheme described below deals with the incoming handoff traffic to the cell as a new traffic.

The general steps taken by the AC scheme are as follows (new terms like *outage probability* and *virtual bandwidth* are described in the next section):

AC-1. A mobile host sends admission request to the base station, together with its desired virtual bandwidth (VB), power limit, and outage probability constraint. As described in the next section, the desired virtual bandwidth is estimated from the two QoS measures mentioned above.

AC-2. The desired VB of the new request is considered with the VBs of the existing traffic.

AC-3. The base station measures the statistics of background interference and path loss from the user.

AC-4. Using the above information, the base station calculates the user outage probability. If a certain outage requirement is not satisfied, the request is denied.

AC-5. If the outage requirement for the requesting user is achievable, the base station further checks whether there is any violation of outage requirements for existing users if the requesting user is admitted. If none is violated, the user is admitted. Otherwise, the request is rejected.

We now introduce the basic notation, definitions, and results used in [24] and [25] to develop the AC scheme, followed by an outline of the important calculation steps.

3.1.1 Notations, Definitions, and Basic Observations

The AC algorithm makes use of the fixed and random quantities listed below (most random variables appear in bold face). The sources of randomness include: user transmission activity (the fraction of time the user is transmitting, described by the *channel-on* probability $p_{i,on}$ below), the outage probability which depends on the users transmission activity and mobility, and the path loss. The analysis given below takes the user transmission activity as the prime source of randomness. The subscript i refers to the ith mobile user/host.

Fixed parameters:

- N: the number of active in-cell users
- P_i: the received power at the serving base station when the user is transmitting
- $P^T_{i,max}$: the maximum transmission power of the mobile device
- R_i: the assigned transmission rate of user i
- SIR_i: the desired SIR (Note: in [24], SIR denotes the quality $(\frac{E_b}{I_0})$ of chapter 2)

- W: the system bandwidth, given by the spreading chip rate in unit of chips per second (cps)

Random Variables

- α_i: a Bernoulli (on-off) random variable characterizing the transmission of user i: $\alpha_i = 1$ when the user is transmitting, and $\alpha_i = 0$ when the user is idle. $p_{i,on}$ denotes the channel-on probability, and $p_{i,off} = 1 - p_{i,on}$ denotes the channel-off probability. Thus, the mean $E[\alpha_i] = p_{i,on}$, and the variance $Var[\alpha_i] = p_{i,on}(1 - p_{i,on})$.
- $\mathbf{L_i}$: the path loss from user i to the serving base station
- $\mathbf{P_i}$: the received power (random variable) at the serving base station: $\mathbf{P_i} = \alpha_i P_i$
- $\mathbf{P_\Sigma}$: the total received in-cell user power ($\mathbf{P_\Sigma} = \sum_{i-1}^{N} \mathbf{P_i} = \sum_{i=1}^{N} \alpha_i P_i$)
- $\mathbf{SIR_i}$: the received SIR at the base station ($\mathbf{SIR_i}$ is a random variable due to the outage probability discussed below)
- $\mathbf{I}$: interference power from user transmission in other cells (commonly treated as as a Gaussian random variable)
- $\mathbf{n_0}$: background noise power (AWGN)
- $\mathbf{I_0}$: is the total background interference $\mathbf{I_0} = \mathbf{I} + \mathbf{n_0}$ (Gaussian)

The development of the AC algorithm uses also the following concepts and observations.

A. Outage Probability. The ith user is at *outage* if $\mathbf{SIR_i} < SIR_i$. The discussion given below identifies a specific quantity that serves as an upper bound, denoted P_{outage}, on the actual outage probability of a user. If such an upper bound P_{outage} is known then one can obtain an upper bound on the packet loss rate (PLR) that may result if the bit error rate (BER) is known as follows:

- If each packet has n bits, and the user is not in an outage state, then the corresponding PLR, denoted PLR_{no} is given by

$$PLR_{no} = 1 - (1 - BER)^n. \tag{3.1}$$

- Furthermore, if an upper bound P_{outage} is known, and assuming that all packets are lost when the user is in an outage state, then an upper bound on the overall PLR is given by

$$PLR = PLR_{no}(1 - P_{outage}) + P_{outage}. \tag{3.2}$$

In section 3.2, the above relation is used to relate the user-defined QoS measure P_L^i to the desired signal-to-interference ratio SIR_i.

B. Virtual Bandwidth (VB). Shen developed the concepts of the *desired* virtual bandwidth, and the *achieved* virtual bandwidth to handle the soft capacity of the CDMA environment. The definition (given below) for the desired VB is not intuitive: at a first look, it just appears as an expression that combines the on-off activity level α_i of the user, the user required rate R_i, the user desired signal-to-interference ratio SIR_i, and the available system bandwidth W to quantify the user needs of the bandwidth. Its main advantage, however, is that it leads to a simplified view of how bandwidth is distributed among different users whose requirements affect each other. In particular:

Definition (adapted from [24]). The *desired virtual bandwidth* of user i, denoted $\mathbf{SR_i}$, is defined as $\mathbf{SR_i} = \alpha_i SR_i$ where SR_i is defined as:

$$\frac{1}{SR_i} \stackrel{\text{def}}{=} \frac{1}{SIR_i.R_i} + \frac{1}{W}.$$

Or, equivalently, $SR_i = \frac{SIR_i.R_i}{1+\frac{SIR_i.R_i}{W}}$.

Some remarks on the above definition are given below.

1. When the user required rate $R_i = 0$, we have $SR_i = 0$, and the desired virtual bandwidth $\mathbf{SR_i} = 0$; this agrees with the intuition of using the bandwidth concept.

2. When $SIR_i.R_i \ll W$ then $SR_i \approx SIR_i.R_i$. For example, when $R_i = 64$ Kbps, $SIR_i = 2$, and $W = 10$ Mcps the quantity SR_i is about 1% of W.

3. The desired virtual bandwidth is defined with respect to the desired signal-to-interference ratio SIR_i. [24] also defines the achieved virtual bandwidth by replacing SIR_i in the above definition with $\mathbf{SIR_i}$ (the achieved SIR at the base station). The achieved VB is denoted $\mathbf{sr_i}$. [24] notes that the achieved VB and the desired VB are two different terms. For example, under perfect power control $\mathbf{SIR_i} \leq SIR_i$, and hence $\mathbf{sr_i} \leq \mathbf{SR_i}$. In the interest of making the presentation short, however, we omit relations involving $\mathbf{sr_i}$.

C. The Cell Model. Shen and Ji [25] used the above concepts to prove the following theorem (called the cell model in their work).

Theorem. A sufficient condition for user i not to be in outage is:

$$\sum_k \mathbf{SR_k} + \mathbf{L_i}\frac{\mathbf{SR_i}}{P^T_{i,max}}\mathbf{I_0} \leq W. \tag{3.3}$$

where the summation of the first term includes all active users (including user i). We refer the reader to [24] or [25] for a proof.

A few notes about the above theorem are given below.

1. An important relation that holds in wire-line when a number of traffic sources send their data in parallel is $\sum \mathbf{R_i} \leq C$, where $\mathbf{R_i}$ is the transmission rate of the ith source, and C is the capacity of the shared link. A similar relation can be derived from relation (3.3) under the following settings and assumptions:

   - set $\mathbf{R_i} = \alpha_i R_i$,
   - approximate $SR_i \approx SIR_i R_i$,

- assume that all mobile hosts request the same signal-to-interference ratio SIR_0 (so, $\sum \mathbf{SR_k} = SIR_0 \sum \alpha_k R_k = SIR_0 \sum \mathbf{R_k}$),
- neglect the background noise $\mathbf{I_0}$, and
- set $C = W/SIR_0$

then we get $\sum \mathbf{R_i} \leq C$.

2. In [24] and [25], the condition $\sum_k \mathbf{SR_k} + \frac{\mathbf{SR_i}}{P_i}\mathbf{I_0} < W$ has also been derived as a sufficient condition for the user not to be in outage. This latter condition reflects that power transmissions in the uplink are correlated: that is, the change of the received power from one user can cause other users to adjust their transmission power.

   The proof of the above observation, however, is embedded in the first part of the proof of the above theorem. To see this, we recall that user i is not in outage if

$$SIR_i \leq \frac{W}{R_i} \frac{P_i}{\mathbf{P_\Sigma} - P_i + \mathbf{I_0}}. \tag{3.4}$$

   Or,

$$\frac{\mathbf{P_\Sigma} - P_i + \mathbf{I_0}}{P_i} \leq \frac{W}{SIR_i.R_i}, \text{ or}$$

$$\begin{aligned} \frac{\mathbf{P_\Sigma} + \mathbf{I_0}}{P_i} &\leq \frac{W}{SIR_i.R_i} + 1 \\ &= W\left(\frac{1}{SIR_i.R_i} + \frac{1}{W}\right) \\ &= \frac{W}{SR_i}. \end{aligned}$$

   When the traffic load is high the above sufficient condition holds as an (approximate) equality; this yields a requirement that $\frac{\mathbf{P_\Sigma}+\mathbf{I_0}}{P_i} = \frac{W}{SR_i}$ must be satisfied. Hence, an increase in the total interference power $\mathbf{P_\Sigma} + \mathbf{I_0}$ requires an increase in user P_i to maintain the same SR_i value.

D. The Definition of P_{outage}. The above theorem allows the following definition of an

upper bound P_{outage} on the outage probability (the definition excludes the case when user i is not transmitting, i.e. when $\mathbf{SR_i} = 0$):

$$P_{outage} \stackrel{\text{def}}{=} \text{Prob}\left\{ W < \sum_{k \neq i} \mathbf{SR_k} + SR_i + \mathbf{L_i} \frac{SR_i}{P^T_{i,max}} \mathbf{I_0} \right\}.$$

Comments:

1. The outage relation defined by the above relation is an individual event for each user. When user i is experiencing outage, other users may (or may not) be experiencing outage, since the term $\mathbf{L_i} \frac{\mathbf{SR_i}}{P^T_{i,max}} \mathbf{I_0}$ is user dependent. That is, step AC-5 has to evaluate the above relation for each user.

2. To evaluate P_{outage} for user i, four quantities are needed:

   - the desired VB information of all users,
   - the path loss $\mathbf{L_i}$,
   - the maximum transmission power limit $P^T_{i,max}$, and
   - the total background interference limit $\mathbf{I_0}$.

   As noted in [24], at the base station, the first quantity is readily available from the requested user rates and SIR values. The path loss can be estimated from the uplink user transmission, such as the pilot signal strength. In addition, users are assumed to report $P^T_{i,max}$ to their serving base stations. Finally, the base station can constantly measure the statistics of the background interference. Hence, the base station can estimate P_{outage} using the above relation.

3.1.2 Details of the Admission Control Algorithm

The following steps detail the computations needed for the AC algorithm.

1. *Calculation of the required transmission rate R_i, and the channel on-off probabilities $p_{i,on}$ and $p_{i,off}$ from the QoS delay constraint D_i.*

   Analysis of this part considers the queueing relation at the user side. In particular,

- packets are assumed to arrive from the user's application to the transmission queue (inside the mobile host) according to a Poisson process with rate λ_i,
- all packets are assumed to have equal lengths of S_i bits; thus, for a fixed assigned transmission rate R_i, each packet requires a constant service time μ_i, and
- packet loss due to buffer overflow, denoted P_B^i, is assumed to be preset to a value much smaller than the required packet loss in the wireless channel P_L^i (e.g., $P_B^i = 10^{-6}$, and $P_L^i = 10^{-2}$).

Under the above assumptions, the user side queue is modelled as an M/D/1 queue, and one can obtain the required service time μ_i.

The required rate R_i can be calculated from $R_i = \mu_i S_i = \lambda_i S_i / \rho_i$, where $\rho_i = \lambda_i / \mu_i$ is the server utilization. Furthermore, when the buffer overflow probability is too small one can approximate the channel-on time by $p_{i,on} \approx \rho_i$, and the channel-off time by $p_{i,off} \approx 1 - \rho_i$.

2. *Calculation of the required SIR_i value, and an upper bound on the outage probability η_i, from the QoS constraint on path loss P_L^i.*

Analysis of this part assumes that lost packets from the mobile host to the base station are not retransmitted (that is, no ARQ protocol is used); the assumption simplifies the analysis.

In equation (3.2), suppose that η_i is an upper bound on the quantity P_{outage} (so, $PLR_{no}(1 - P_{outage}) + P_{outage} \leq PLR_{no}(1 - \eta_i) + \eta_i$), and let P_L^i be the given QoS constraint on the packet loss probability, then the AC algorithm should satisfy the relation:

$$PLR_{no}(1 - \eta_i) + \eta_i \leq P_L^i.$$

Equation (3.1), and the exact relation between the BER and the SIR of the used

receiver structure allow us to write PLR_{no} for the ith user as function of SIR_i; that is, one can write $PLR_{no} = g_i(SIR_i)$, using some suitable function g_i. To satisfy the above relation, we need to solve

$$g_i(SIR_i) \leq \frac{P_L^i - \eta_i}{1 - \eta_i}$$

or

$$SIR_i \geq g_i^{-1}\left(\frac{P_L^i - \eta_i}{1 - \eta_i}\right).$$

The above inequality can be satisfied by choosing suitable values for SIR_i and η_i (various solutions may exist, and one can pick any feasible values).

3. *Approximating the mean m_i, and variance σ_i^2, of $\mathbf{SR_i}$ (the desired VB) from the channel rate R_i (obtained in step 1), and SIR_i (obtained in step 2).*

Analysis in this part assumes that $SR_i \approx SIR_i.R_i$. Or, from the queueing relations mentioned above: $SR_i \approx (SIR_i\lambda_i S_i)/\rho_i$. Denote by the SR_{i0} the constant product $SIR_i\lambda_i S_i$, then $SR_i = SR_{i0}/\rho_i$. $\mathbf{SR_i} = (SR_i 0/\rho_i)\alpha_i$. Hence the mean

$$\begin{aligned} m_i = E[\mathbf{SR_i}] &\approx (SR_{i0}/\rho_i)\, E[\alpha_i] \\ &= SR_{i0}, \end{aligned}$$

and the variance

$$\begin{aligned} \sigma_i^2 = Var[\mathbf{SR_i}] &\approx (SR_{i0}/\rho_i)^2\, Var[\alpha_i] \\ &= (SR_{i0}/\rho_i)^2\, \rho_i(1-\rho_i) \\ &= (\frac{1}{\rho_i} - 1)\, (SR_{i0})^2. \end{aligned}$$

(Note that a small ρ_i results in a large variance of the desired VB.)

4. *Bounding the outage probability P_{outage} using a Gaussian approximation.*

The relevant assumptions in this part are related to the terms in the expression $\text{Prob}\left\{W < \sum_{k\neq i} \mathbf{SR_k} + SR_i + \mathbf{L_i}\frac{SR_i}{P_{i,max}^T}\mathbf{I_0}\right\}$, as stated below.

- The number of active users N at the time of decision making instant is sufficiently large so that $\mathbf{SR_\Sigma^i} = \sum_{k \neq i} \mathbf{SR_k}$ can be approximated by a Gaussian random variable with the following mean and variance:

$$m_\Sigma^i = \sum_{k \neq i} m_k = \sum_{k \neq i} SR_{k0},$$

$$\sigma_{i,\Sigma}^2 = \sum_{k \neq i} \sigma_k^2 = \sum_{k \neq i} (\frac{1}{\rho_k} - 1)(SR_{k0})^2.$$

where $SR_{k0} = SIR_k \lambda_k S_k$.

- The path loss $\mathbf{L_i} = \ell_i$ is known and static. The assumption is applicable to the evaluation of instantaneous outage probability, or when the user is not moving. When a user is moving fast, P_{outage} has to be evaluated constantly. Letting $a_i = \ell_i / P_{i,max}^T$, the term $\mathbf{L_i} \frac{SR_i}{P_{i,max}^T} \mathbf{I_0}$ simplifies to $a_i SR_i \mathbf{I_0}$.
- Furthermore, if the background interference $\mathbf{I_0}$ is treated as Gaussian, with mean m_I and variance σ_I^2 then the random variable $\mathbf{SR_\Sigma^i} + SR_i + a_i R_i \mathbf{I_0}$ can be approximated by a Gaussian random variable with mean

$$m_{all} = m_\Sigma^i + SR_i + a_i SR_i m_I$$

and variance

$$\sigma_{all}^2 = \sigma_{i,\Sigma}^2 + (a_i SR_i \sigma_I)^2.$$

5. Finally, the outage constraint can be expressed using the standard Q-function as

$$Q(\frac{W - m_{all}}{\sigma_{all}}) \leq \eta_i.$$

The above relation is the main admission control condition tested in step AC-4 for the new request, as well as in step AC-5 to check whether there is any violation of outage required for each of the existing users if the requesting user is admitted.

3.1.3 Overview of the Obtained Results

The framework described above has been used to study system capacity aspects in different scenarios. In [28], for example, the authors start from a point where R_i, SIR_i, $p_{i,on}$ and

η_i are explicitly given to the AC algorithm (i.e., the computations start from step 3 of the previous section, and there is no given QoS parameters D_i and P_L^i). By focusing on two types of traffic: type 1 ($R_1 = 64$ Kbps, $SIR_1 = 2$, and $\eta_i = 0.4$), and type 2 ($R_2 = 144$ Kbps, $SIR_2 = 2$, and $\eta_i = 0.2$), and assuming all users require the same upper bound η on their outage probabilities, the authors study the effect of varying η in the range $[0.001, 0.05]$ on the number $N_1 + N_2$ of admitted type-1 and type-2 users. The results in [28] show, for example, that under the constraint of equal received power from all users, and that the background noise $\mathbf{I_0}$ equals a fraction $\gamma \mathbf{P_\Sigma}$, where $\gamma = 1/2$, then the relaxation of η from 0.001 to 0.005 has a dramatic effect in admitting more users (e.g. the number increases by a factor of 4). This effect, however, diminishes as η is further relaxed from 0.005 to 0.05 (where the gain in number of admissible users is just about 20%).

3.1.4 Discussion

The concepts of (achieved and desired) VB offer a formalization that captures the interaction between the basic quantities involved in determining the required SIR at the base station. The concepts do not appear to lead to a reduction in the computational work required to ensure that a new user can be admitted while maintaining the SIR for all existing users. Accounting for user mobility will further increase the computations done by the algorithm. The analysis also does not consider the main point of the thesis which is ensuring proportional delay differentiation. Hence, new approaches have to be developed.

3.2 Work on Weighted Fair Scheduling

Packet scheduling algorithms such as *weighted fair queueing* (WFQ), and other algorithms that approximate the behaviour of the idealized *generalized processor sharing* (GPS) service discipline, are important traffic management techniques for providing QoS in the design of conventional wire-line routers. For a background on the above material, see for example, the chapters on scheduling in [13] or [6], or the survey article of [34].

Applications of approximate GPS algorithms to packet scheduling in wireless environ-

ments has recently started to receive attention. Two recent efforts in this direction are

- the work of [17] on a TDMA system (part of the Ph.D. research of S. Lu), and
- the work of [32] and [31] (part of the Ph.D. work of L. Xu) on the uplink traffic in W-CDMA. The main results of this work are two algorithms:
  - **CDGPS** (for *code-division GPS*): an algorithm that serves each traffic flow during any time slot so as to satisfy certain weighted fair scheduling constraints.
  - **A-CDGPS** (for *aggressive CDGPS*): an improved algorithm that allocates less than the fair rate to flows that are not sufficiently backlogged to warrant the allocation of their corresponding fair rates.

In this section, we outline the above two algorithms. To introduce the material we find it necessary to first outline some background information on GPS (Section 3.2.1). Then we outline some basic assumptions and observations about the CDGPS and A-CDGPS in section 3.2.2. Sections 3.2.3 and 3.2.4 discuss some detailed aspects of the algorithms. Section 3.2.5 summarizes the main findings in [31]. Finally, we conclude with some remarks in Section 3.2.6.

3.2.1 Background on GPS

We recall from [21] the following basic aspects of GPS.

1. The GPS service discipline is an idealized service discipline that approximates a traffic flow as a continuous fluid flow. That is, it assumes that during any arbitrarily small period of time an arbitrarily small amount of data can be served.
2. A traffic flow (or session) is called *backlogged* at time t if a positive amount of that session's traffic is queued at time t.
3. A GPS server that serves N flows (or sessions) is characterized by a set of N positive real numbers $\phi_1, \phi_2, \cdots, \phi_N$ that denote the relative amount of service that should be given to each flow. In particular, if $S_i(\tau, t)$ denotes the amount of flow i traffic

served during the interval $[\tau, t]$ then the following constraints should be satisfied for any flow that is continuously backlogged in the interval $[\tau, t]$:

$$\frac{S_i(\tau,t)}{S_j(\tau,t)} \geq \frac{\phi_i}{\phi_j}, \quad j = 1, 2, \cdots, N \tag{3.5}$$

4. Summing the above inequalities over all flows j we get:

$$S_i(\tau,t) \sum_{j=1}^{N} \phi_j \;\geq\; \phi_i \sum_{j=1}^{N} S_j(\tau,t).$$

If we denote by r the total rate available to the server, then the amount of data served during the time interval $t - \tau$ is: $\sum_{j=1}^{N} S_j(\tau,t) = (t-\tau)r$, and the above relation simplifies to $S_i(\tau,t) \sum_{j=1}^{N} \phi_j \;\geq\; \phi_i(t-\tau)r$.

5. The above relation implies that flow i is guaranteed a rate of $g_i = \frac{S_i(\tau,t)}{t-\tau} \;\geq\; \frac{\phi_i}{\sum_j \phi_j} r$.

6. Under GPS, performance bounds can be shown for *leaky bucket* constrained flows. A leaky bucket traffic regulator (or shaper) for flow i is characterized by two parameters: a token bucket of size σ_i, and a token arrival rate of ρ_i. The flow conforms to the leaky bucket constraints if the amount $A_i(\tau,t)$ of session i traffic entering the network during any time interval $[\tau, t]$, $t - \tau \geq 0$, satisfies:

$$A_i(\tau,t) \leq \sigma_i + \rho_i(t-\tau). \tag{3.6}$$

3.2.2 Basic Assumptions and Observations

The following collection of assumptions, definitions, and basic observations are needed to present the algorithms.

1. The positive weights $(\phi_i |\; i = 1, 2, \cdots, N)$ are assumed to be known to the base station. For example, the network provider may associate a certain weight with each type of traffic flows (for example, $\phi_{voice} = 8$, $\phi_{video} = 260$, and $\phi_{bestEffort} = 2$, and so on).

2. Both of the CDGPS and the A-CDGPS algorithms use the following general framework.

- Time is slotted. Each time slot has length T.
- By the end of each time slot, each mobile host sends a short message (through a random channel or a dedicated channel) reporting the buffer state, and the amount of data received during the previous slot.
- Upon receiving bandwidth requirements from all active mobile users, the scheduler located in BS allocates the channel rate for the next slot, taking into account user QoS' requirements and the available uplink capacity.
- A resource allocation message including channel rates will be sent through a downlink channel.
- Upon receiving the above message, the mobile hosts adjust their rates, and start transmitting data.

3. As in Chapter 2, each traffic flow i has a required target $(E_b/I_0)_i$ ratio for which the following inequality should be satisfied:

$$(E_b/I_0)_i \leq \frac{W}{R_i} \frac{P_i}{P_\Sigma - P_i + I_{ext}}.$$

Now, let $\gamma_i = \frac{P_i}{P_\Sigma - P_i + I_{ext}}$ be the minimum signal-to-interference ratio required by flow i, and consider the fraction $\frac{\gamma_i}{1+\gamma_i}$. Note that:

- Since, $\frac{\gamma_i}{1+\gamma_i} = \frac{P_i}{P_\Sigma + I_{ext}}$, then summing over all N active flows, one obtains the following necessary condition for the flows to be admissible:

$$\sum_{i=1}^{N} \frac{\gamma_i}{1+\gamma_i} = \frac{P_\Sigma}{P_\Sigma + I_{ext}} \quad (\leq 1).$$

(Note: when the mobile host uses the minimum amount of power to achieve the required $(E_b/I_0)_i$ ratio, the quantity $\frac{\gamma_i}{1+\gamma_i}$ used here becomes equivalent to the quantity $C_i = \frac{1}{\frac{W}{(E_b/I_0)_i R_i}+1}$ used in Section 2.5.)

- From the above inequality, the quantity $\frac{\gamma_i}{1+\gamma_i}$ can be used as a measure of the resources given to the ith flow. (Note also that the larger γ_i gets, the larger the fraction $\frac{\gamma_i}{1+\gamma_i}$ becomes.)

- Furthermore, if we let $\eta = \frac{I_{ext}}{P_\Sigma}$ then the above condition simplifies to $\sum_{i=1}^{N} \frac{\gamma_i}{1+\gamma_i} = \frac{1}{1+\eta}$.
- Rewriting $\frac{1}{1+\eta}$ as $1-\delta$ (where $\delta = \frac{\eta}{1+\eta}$), one can express the above necessary condition in the simple form:

$$\sum_{i=1}^{N} \frac{\gamma_i}{1+\gamma_i} = 1 - \delta.$$

- In [31] performance is studied assuming that $\eta = 0.5$, and hence $\sigma = 0.33$.

4. If C denotes the available uplink bandwidth, and we want to allocate a rate $g_i = \frac{\phi_i}{\sum_{j=1}^{N} \phi_j} C$ to the ith flow, $i = 1, 2, \cdots, N$, then user i share of the signal-to-interface ratio is $\gamma_i = \frac{g_i}{W}(\frac{E_b}{I_0})_i$, and C can be determined by finding the positive solution to the equation

$$\sum_{i=1}^{N} \frac{\gamma_i}{1+\gamma_i} = 1 - \delta. \tag{3.7}$$

5. Let $S_i(k)$ be the amount of flow i traffic that would be served during time slot k, then the CDGPS algorithm is designed to satisfy the following weighted fair scheduling constraints: for any flow i that is continuously backlogged during time slot k we have

$$\frac{S_i(k)}{S_j(k)} \geq \frac{\phi_i}{\phi_j}, \quad j = 1, 2, \cdots, N.$$

3.2.3 The CDGPS Algorithm

At the start of time slot k, the algorithm computes the available capacity C, as discussed for relation (3.7) above. Then for each flow i, the algorithm computes the minimum guaranteed rate: $g_i = \frac{\phi_i}{\sum_{j=1}^{N} \phi_j} C$. Then it executes the following steps.

1. For each flow i, estimate the amount $b_i(k)$ of the available data in the mobile host's buffer. This can be done based on the amount of backlogged traffic at the end of the previous slot, and the arrival rate of flow i.

2. Based on the estimated $b_i(k)$, $i = 1, 2, \cdots, N$, determine the weighted fair amount $S_i(k)$ of traffic to be served (recall that each time slot has length T) as follows.

(a) If $b_i(k) = 0$ then set $S_i(k) = 0$.

(b) If $b_i(k) > 0$ then set $S_i(k) = g_i T$.

(c) If $\sum S_i(k) < CT$, then distribute the remaining resources among the users in proportion of the weights according to the GPS service discipline.

3. The allocated channel rate to user i is determined by $R_i = S_i(k)/T$.

The following delay bound is mentioned in [31].

Theorem [31]. Let flow i be regulated by a leaky bucket with token buffer size σ_i and token generation rate ρ_i. If $g_i \geq \rho_i$ then the packet delay of flow i is bounded by

$$Delay_{max} \leq \frac{\sigma_i}{g_i} + T.$$

3.2.4 The A-CDGPS Algorithm

The A-CDGPS algorithm improves on the above algorithm by exploiting the following aspect: at the start of any time slot k, if the estimated amount $b_i(k)$ of flow i traffic is not large enough to warrant the allocation of the "guaranteed minimum fair rate" g_i (i.e., $b_i(k) << g_i T$) then one can allocate the smaller rate $\frac{b_i(k)}{T}$ to flow i, and use the remaining bandwidth to serve other backlogged flows.

One interesting question is how to implement the above aspect. The method used in [31] is based on the following steps. (Note: the simplified presentation given below is not exactly the presentation given in [31].)

1. First, compute the available capacity C by finding a positive solution to relation (3.7) above. Then compute the fair rate g_i for each flow i.

2. Let X be the set of flow indexes where flow $i \in X$ needs less than its fair share of the capacity (that is, $b_i(k) << g_i T$, as mentioned above).

3. For each flow $i \in X$, assign the reduced rate $\frac{b_i(k)}{T}$, and compute the associated (reduced) signal-to-interference ratio $\gamma_i = \frac{b_i(k)}{TW} (\frac{E_b}{I_0})_i$.

4. Let $\delta_X = \sum_{i \in X} \frac{\gamma_i}{1+\gamma_i}$. Note that allocating the signal-to interference ratio γ_i to each flow $i \in X$ reduces the available capacity from $1 - \delta$ to $1 - \delta - \delta_X$.

5. Denote by C' the remaining available capacity. For flows in the complement set $\overline{X}$ (flows in $\overline{X}$ have not been assigned rates yet), let $\phi_{\overline{X}} = \sum_{i \in \overline{X}} \phi_i$. Let $R'_i = \frac{\phi_i}{\phi_{\overline{X}}} C'$, and $\gamma'_i = \frac{R'_i}{W} (\frac{E_b}{I_0})_i$. Compute C' by finding a positive solution to the following equation:

$$\sum_{i \in \overline{X}} \frac{\gamma'_i}{1+\gamma'_i} = 1 - \delta - \delta_X.$$

6. For each mobile $i \in \overline{X}$, allocate the rate $R'_i = \frac{W\gamma'_i}{(E_b/I_0)_i}$.

3.2.5 Overview of the Results

The simulation results in [31] show the achieved delay and utilization of the two above algorithms under a mix of heterogeneous traffic flows described below.

- Ten voice flows, each voice flow is generated by using an ON-OFF model, where the activity factor is 0.4, and in the ON-state the packets are generated at a constant rate $R_{voice} = 8$ Kbps (= 100 packets/second with a packet length = 80 bits).

- One VBR video traffic flow, generated using an 8 state MMPP model and a leaky bucket regulator. The average duration in each state is 40 msec. The maximum rate of the VBR video is 384 Kbps, and the average rate is $\rho_{video} = 180$ Kbps. The leaky bucket parameters are $(\rho_{video}, \sigma_{video})$, where $\sigma_{video} = 18$ Kbits.

- Four best-effort flows, each flow is generated by a Poisson process with packet size of 1600 bits, and an average arrival rate of 50 Kbps.

- The A-CDGPS algorithm (but not the CDGPS algorithm) uses the following weights: $\phi_{voice} = 8$, $\phi_{video} = 260$, and $\phi_{bestEffort} = 2$.

In the simulation setup:

- each time slot $T = 10$ msec,

- a conservative lower bound C_0 on the available capacity is obtained by solving equation (3.7) assuming all flows require the same $(E_b/I_0) = 10$ dB, and all flows are assigned equal weights (so, $\phi_i = 1/N$, $N = 15$),
- the conservative lower bound C_0 is used for all iterations of the CDGPS algorithm, and
- the average (over all flows) packet arrival rate normalized by C_0 is taken as a measure of the traffic load.

Under the above conditions, the results show that the A-CDGPS outperforms the CDGPS algorithm.

3.2.6 Discussion

Based on the above information, one can make the following remarks.

1. The above algorithms do not serve the purpose of achieving proportional delay differentiation between aggregate classes of traffic. Hence new approaches should be developed.
2. The above GPS algorithms guarantee delay bounds on a session's flow only if the flow conforms to the parameters of a leaky bucket regulator, and the cell load allows the server to allocate bandwidth more than the token arrival rate parameter. Under such circumstances, each packet incurs a rather small delay.
3. Since GPS-based algorithms provide a tool to achieve service differentiation, it is possible that similar ideas can be used to provide proportional delay differentiation; this hypothesis, however, remains a subject of future research.

Chapter 4

Simulation Environment

The previous chapter has outlined some important aspects of power allocation in the uplink of cellular CDMA networks. In this chapter, we discuss the simulation environment used to evaluate the performance of the PDD schedulers and CAC algorithms proposed in the next chapters. Some parameters included in the simulation are adopted from the 3GPP standards [3]. We organize this chapter as followes: Section 4.1 gives us an overview of the communication protocol used in our simulation environment. Section 4.2 describes the simulation parameters used in our study.

4.1 Overview of the Communication Protocol

As in the algorithms presented in chapter 3, time is considered to be slotted. The power control and transmission rate assignment done by the proposed scheduling algorithms are assumed to be done in a synchronous fashion with a fixed time interval. We call the time interval between two consecutive transmission rate assignments a *time control slot* and denote its length as T (ms) (see Figure 4.1). In our simulation, T = 500 ms.

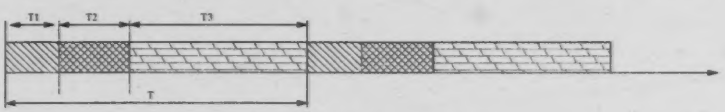

Figure 4.1: Control Slot Model

Each control slot is divided into three parts: $T1$, $T2$, and $T3$.

- During the time period $T1$, the base station collects status information from active mobiles by allowing mobiles sending a short message to the BS through some special channels. The status information includes the arrival time of packets, packet lengths, and the amount of traffic that arrived in the previous time slot.

- During $T2$, the BS first executes a CAC algorithm to decide which mobiles are admitted into the system. Subsequently, the BS executes a PDD scheduler, and inform the mobiles with the assigned rates. Upon receiving the allocated rates, each MS immediately adjusts its spreading factor before $T3$.

- During $T3$, the active mobiles transmit data packets to the BS with the newly assigned transmission rate.

We assume that each mobile has one queue that holds a flow that belongs to one particular DiffServ class.

4.2 Simulation Parameters

Our simulation experiments are based on snapshots where mobile users are uniformly distributed in a target cell.

4.2.1 User Mobility Parameters

Figure 4.2 illustrates a typical 19-cell configuration where the base stations are placed on a hexagonal grid with distance of 1000 meters apart, and the radius of each cell therefore is equal to 577 meters. In our experiments, the interference caused by neighbouring cells are accounted for by adjusting the ratio $f(t) = \frac{I_{ext}}{P_\Sigma}$ in equation (2.7)

Mobile devices are uniformly distributed in the cell at the beginning of each simulation run. Thereafter, each mobile will roam around in the cell at a certain speed, like 15 m/s. Since we do not consider the handover case, we assume that mobile hosts only move within the cell. We consider a random mobility model where mobiles choose one out of eight possible directions randomly to move toward every 1 second. The movement direction is

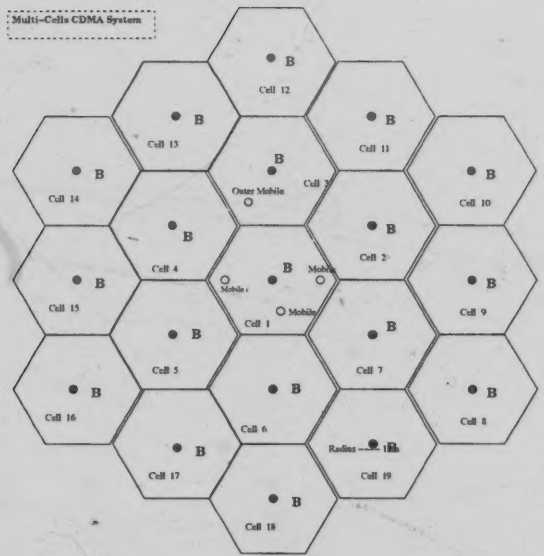

Figure 4.2: Multi-cell environment

Mobility Parameters	*Value*	*Unit*
Maximum Mobiles in Simulation	20	
Mobile Speed	15	m/s
Mobility Update Interval	1	*second*
Mobility Direction	8	

Table 4.1: Mobility Parameters

one element of the set given by:

$$direction \in [0, \frac{\pi}{8}, \frac{2\pi}{8}, \frac{3\pi}{8}, \frac{4\pi}{8}, \frac{5\pi}{8}, \frac{6\pi}{8}, \frac{7\pi}{8}].$$

Mobility parameters are summarized in table 4.1.

4.2.2 CDMA paramters

In our simulation, we assume that the chip rate $W = 4.096$ Mcps, and the background noise contained in W is also a common parameter, $3.98e-13$ Watt. In all of the simulation runs, target bit-energy to interference-power ratio ($\frac{E_b}{I_0}$) is 7 (dB). To simplify the problem, we assume that all mobiles require the same $\frac{E_b}{I_0}$ ratio. Other parameters about link and propagation model are summarized in table 4.2.

CDMA Parameters	*Value*	*Unit*
Chip Rate	4.096	*Mcps*
Target $\frac{E_b}{I_o}$	7	*dB*
Noise Density Power	-168	*dBm*
Max Transmission Power	700	*mWatt*
Transmission Rate	$16 - 192$	*Kbps*
Convolutional coding rate	1/3	
Log − normal shadowing standard deviation	4	*dB*
Power update interval	500	*ms*
Total # of cells	19	
Cell Radius	577	*Meter*

Table 4.2: CDMA Parameters

4.2.3 Traffic Parameters

To demonstrate the performance of the proposed PDD scheduler and CAC methods for heterogeneous traffic, three priority classes with different weights are used. Mobiles are classified into these 3 classes according to their QoS requirements. We assume that each class counts 1/3 of the total system traffic load.

In our simulation work, the traffic parameters do not reflect the use of any particular user application, rather the setting of the parameters is intended to generate a spectrum of traffic loads ranging from light load to heavy load. The wireless system can support multiple transmission rates from 8 Kbps for voice up to 2 Mbps (for a single user) for data applications. We choose a range of transmission rates from 16 Kbps up to 192 Kbps in order to drive the system from non-congested to congested states so as to examine our approaches.

Once a mobile is admitted into the system, it starts generating flows of packets. In all cases we assume:

- the packet interarrival time follows an exponential distribution with a mean of 0.5 sec.
- The packet length follows a discrete distribution where 45% of the packets are 120 bytes, 40% of packets are 550 bytes, and 15% of the packets are 1500 bytes. This

Traffic Parameters	*Value*	*Unit*
Number of Priority Classes #	3	
Number of Queue in Each Mobile	1	
Packet Length Dist.	$45\% - 120, 40\% - 550, 15\% - 1500$	*Bytes*
Packet Time Last	6	*second*
Mean Packet Interarrival Time	0.5	*second*

Table 4.3: Traffic Parameters

packet distribution applies to all classes.

- A packet is dropped if it stays in a queue longer than 6 sec.

The above packet distribution are similar to distributions used in many existing studies [8]. Traffic parameters are summarized in table 4.3

4.2.4 Setting the Maximum Number of Mobiles

In our simulation experiments, traffic is generated by N mobile users. A generated traffic has an equal probability of requesting one of the three DiffServ classes. If N is set too small, the generated traffic load will be very small. If N is set too large, then the call admission control will reject many requests, and the simulation time will be wasted. In this section, we set N so that:

- the cell will have a light traffic load if each mobile is served at a rate R in the range 16 - 64 kbps.
- the cell will have a moderate traffic load if each mobile is served at a rate R in the range 128 - 176 kbps.
- the cell will have a heavy traffic load if each mobile is served at a rate R above 176 kbps.

$$N_{max} = \left(\frac{\frac{W}{R} + 1}{1 + f(t)} \right) \tag{4.1}$$

$Trans\ Rate\ R\ (Kbps)$	$Spreading\ factor\ \frac{W}{R}$	N_{max}	$N_{max}^{'}(\frac{N_{max}}{0.5})$
32	128	20	40
64	64	11	22
128	32	6	12

Table 4.4: Capacity for different spreading factor

Column 3 of table 4.4 is computed from equation (4.1) above assuming the required $\frac{E_b}{I_0}$ value $\zeta = 7$ dB for all mobiles, $f(t) = 0.3$, and each active mobile is continuously transmitting all the time. Column 4 of table 4.4 assumes a 50% activity factor for each mobile. As can be seen, setting N_{max} to 20 is suitable to fulfill the above requirements.

Chapter 5

A Basic Proportional Delay Differentiation Scheduler

We have presented our simulation model and the related parameters in the previous chapter. In this chapter, we devise a basic PDD scheduling algorithm and a basic call admisstion control that takes into account the soft capacity aspect of the system. In addition, simulation results based on these algorithms will be illustrated. In particular, Section 5.1 gives a background on PDD, and presents a basic scheduler. Section 5.2 demonstrates the simulation results of the basic PDD scheduling algorithm. Section 5.3 outlines the devised basic call admission control algorithm. Section 5.4 compares the performance of a system with and without a CAC scheme.

5.1 Background

As mentioned in chapter 1, in a proportional delay differentiation (PDD) architecture the traffic is classified into a small number K of delay classes. At the wireless-Internet gateway the ith delay class is associated with a delay differentiation parameter Δ_i (defined by the network provider). In our presentation, class 1 has the lowest priority, class K has the highest priority, and $\Delta_1 > \Delta_2 > \cdots > \Delta_K$. An ideal PDD scheduler should satisfy the following relation: if $\overline{d_i}(t)$ denotes the average delay incurred by class i packets over a time window of length t during which the classes are continuously backlogged, then for any two

classes i and j

$$\frac{\overline{d_i}(t)}{\overline{d_j}(t)} = \frac{\Delta_i}{\Delta_j}.$$

For example, if $\Delta_1 = 4$, $\Delta_2 = 2$, and $\Delta_3 = 1$ then, ideally, class-1 packets should wait on the average four times as long as class-3 packets.

Methods for providing such delay differentiation have been discussed, for example, in [8, 15, 18] for wireline routers, and [30] for the downlink of CDMA cellular wireless networks. We note the following:

1. The DiffServ architecture aims at achieving a PHB over a sufficiently long periods of time. So, the window length t mentioned above is on the scale of multiple sessions (not any arbitrarily small length of time).

2. The condition that during t sec the sessions are continuously backlogged is required since, otherwise, one of the classes may have a zero average delay because it did not have packets to serve.

3. In [30], the following basic difference between the PDD scheduling in wireline and wireless CDMA environments has been noted: in the wireline case one can assign one queue to all flows belonging to each delay class. In each time slot, as many packets are transmitted from each queue (in a first-come first-served order) to satisfy the required relative delay constraints. Many packets may belong to the same flow. In the wireless environments, however, time is slotted, and there is an upper bound on the traffic amount that can be received from any single user during any time slot. Moreover, parallel concurrent transmissions from different mobiles should take place during any slot.

The proposed scheduler discussed here takes the above aspect into consideration by keeping a queue at the base station (or, more accurately, at some wireless packet-level controller module) for each active flow within each delay class, as shown in Figure 5.1 below.

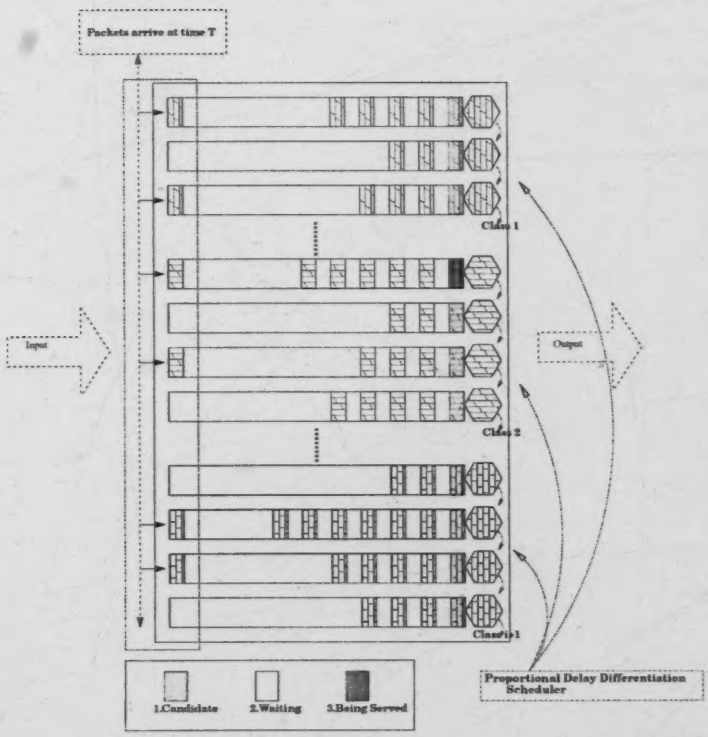

Figure 5.1: Queueing model for Proportional Delay Differentiation scheduling

5.1.1 A Basic PDD Scheduler

A basic approach that underlies many of the methods devised in [8, 15, 18, 30] to construct a PDD scheduler is to compute a *normalized delay* (a priority measure) for each class-i packet (or for an average delay of a subset of class-i packets). The normalized delay (priority) increases in proportion of both the inverse weight $1/\Delta_i$, and the time the packet spends waiting for service. In a wireless environment, such packet priorities are computed at the beginning of every scheduler control slot, and the resulting values are used to perform scheduling during the remainder of the slot. More specifically, if

- p_{ij}^k is a class-i mobile-j packet considered for transmission in slot k,
- τ_{ij}^k is the arrival time of packet p_{ij}^k,
- t is the current time, and
- $D(p_{ij}^k, t)$ is the waiting time $(t - \tau_{ij}^k)$

then the normalized delay (or priority) of packet p_{ij}^k is defined as $P(p_{ij}^k, t) = \frac{D(p_{ij}^k, t)}{\Delta_i}$.

Following the same principle, we design a basic PDD scheduler that works as follows. During any time slot k, the scheduler tries to serve a number of mobile users by allowing each user to transmit at a predetermined rate R (the rate R varies from 16 Kbps to 192 Kbps). At the end of the previous $k-1$ slot, the head of queue in each mobile may store a partial packet (that could not be completely transmitted during slot $k-1$). The following steps are then executed during slot k.

1. During sub-slot T_1, each mobile sends a vector of current packet (or partial packet, as mentioned above) delays, and the associated packet length to the base station. The vector is in decreasing order of the delays. Information about as many packets as the length of sub-slot T_1 permits is transmitted.

2. For each user, the controller computes a vector of normalized delays (priorities) from the received vector of packet delays. Each normalized delay vector is stored in a decreasing order in a user-specific queue.

3. During sub-slot T_2, the controller chooses packets for subsequent transmission during the remaining period of slot k. Packets are considered for transmission in decreasing order of their normalized delays (priorities). A packet p_{ij}^k with the highest normalized delay is admitted during slot k if it satisfies the following conditions, otherwise, the scheduler skips the packet to the next highest priority packet. The conditions are:

   (a) Mobile j can transmit (at rate R) during slot k, along with all other mobiles chosen to be active.

   (b) The total amount of traffic scheduled for transmission by mobile j does not exceed $R.T_3$ bits.

Algorithm 1 summarizes the procedure one step by step.

Algorithm 1 Scheduling algorithm based on TDP

parameters:
S**: a set of selected packets**
Δ_i**: proportional control parameters**
N**: maximum number of active mobiles that the system can support**

algorithm:
$S = \emptyset$;
while (there are queued packets that have not been assigned priority) **do**
  for (mobile $j = 1$ to N) **do**
    Calculate delay of the first packet in the queue of each mobile j
    Calculate the priority for each head packet based on formula:
      $P(p^k{}_{ij}, t) \equiv \frac{D(p^k{}_{i,j}, t)}{\Delta_i}$
    Add these packets into the selected packets set S, sort them from highest to lowest
    Exclude the packets that have been put into S from re-calculating priority
  end for
end while
Decide the number of packets in set S such that the amount of bits is $\leq R \cdot T_3$
Inform mobiles to transmit the chosen packet in set S using allocated transmission rate

5.2 Simulation Results

In this section, we investigate the proportional delay differentiation performance without call admission control under the three different system congestion levels in terms of the

achieved average class delay, delay ratios among classes, and effective throughput. We examine whether the proportional delay differentiation can achieve the target proportional delays among different classes at moderate congestion level, and its performance under heavy system load.

5.2.1 Average Class Delay

From Figure 5.2, when the system is in the range of data rates [144 - 192] kbps, we observe that Class 1's delay is mostly in the range of [2000 - 3500] msec, Class 2's delay is mostly in the range of [900 - 1600] msec, and Class 3's delay is mostly in the range of [400 - 600] msec. According to the delay control parameters of 4:2:1, this result is close to what we expect. Class 1's delay is approximately double of Class 2's delay, which is in turn approximately double of Class 3's delay. The expected proportional delay among classes is achieved at 128 Kbps transmission rate when we take a detailed look during the simulation.

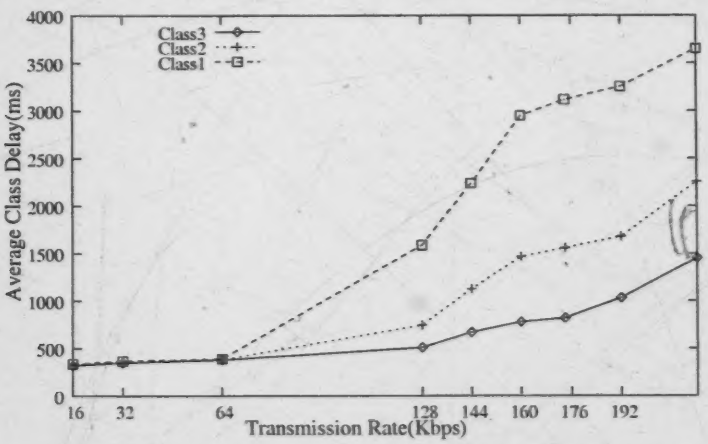

Figure 5.2: Average Class Delay Without CAC

5.2.2 Average Class Delay Ratio

Figure 5.3 illustrates the average delay ratio achieved under different data transmission rates, which will create different congestion levels. In Figure 5.3, we explore how delay differentiation scheme performs under different system congestion. The solid line repre-

sents Class 3 to Class 1 delay ratio, which is expected to be around 4. The dotted line is Class 2 to Class 1 delay ratio, which is expected to be around 2.

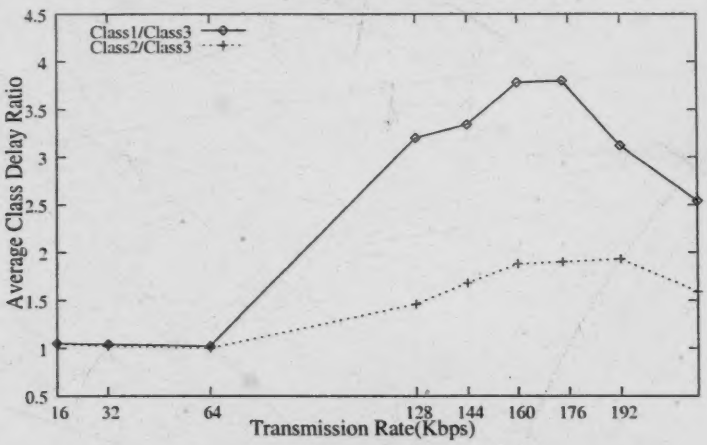

Figure 5.3: Average Class Delay Ratio Without CAC

We observe that each curve in Figure 5.3 is above 1. Also, the Class 1 to Class 3 delay ratio curve is always above the Class 2 to Class 3 delay ratio curve for each rate. Thus, for each transmission rate, proportional delay differentiation scheduling algorithm guarantees that Class 3's delay is always lower than Class 2's delay, which is in turn lower than Class 1's delay. In general, a higher priority class is guaranteed a lower average class delay.

There exist three regions of different ratios in Figure 5.3:

- The light traffic region ranges from 16 Kbps to 64 Kbps.
- The moderate region is from 128 Kbps to 176 Kbps.
- The heavy traffic region is above 176 Kbps.

These three regions of different delay ratio characteristics are a result of different congestion levels caused by different transmission rates.

5.2.3 Average Class Throughput

In Figure 5.4, we observe that, from 16 Kbps to 64 Kbps, almost all of the packets arrived at the system are transmitted. The simulation result shows that no packet is dropped due to

delay in this rate region. Thus, the throughput does not make big difference for different classes .

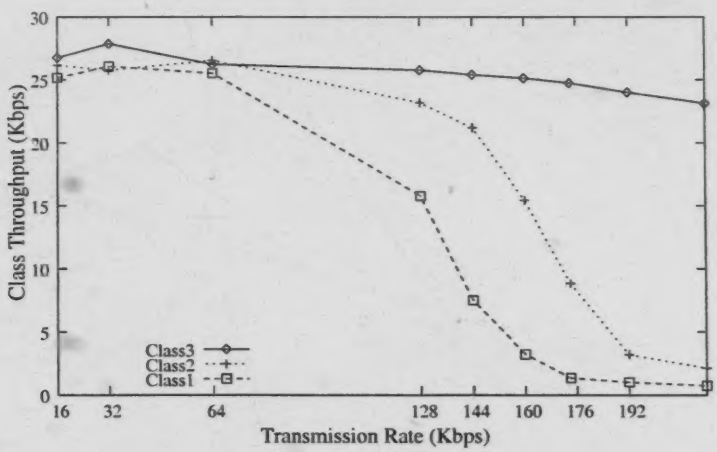

Figure 5.4: Average Class Throughput Without CAC

However, for 128 Kbps to 192 Kbps, the system capacity becomes lower because of interference. The scheduling algorithm determines the transmission priority among different classes and, in turn, the throughput according to the control parameters. We observe that the highest priority class (class 3) maintains approximately the same level in the interval [128 Kbps, 192 Kbps] while the throughput of class 2 and class 1 decreases significantly. As the transmission rate increases, most of Class 1 and Class 2's packets can not get service. Without admission control, all traffic is competing for the limited bandwidth. Under these circumstances, when performing delay differentiation scheduling, a higher class is guaranteed a lower delay at the cost of lower classes' starving. Throughput unfairness among classes occurs when the system is congested, and this unfairness is more notable when the system is heavily congested.

5.3 A Basic Call Admission Control Scheme

In order to alleviate the above problems, we present a call admission control algorithm embedded as part of the resource manager that executes in a wireless-Interent gateway.

The CAC is responsible for regulating the admission of flows that will go from the mobile hosts to the base station to the wireless-Internet gateway, and subsequently to other hosts (if accepted by the DiffServ bandwidth broker in change of the DS domain in which the source host lies).

In our design, we assume that the bottleneck is in the communication between the mobile hosts and the base station.

- At the beginning of each $T1$ control slot, the wireless-Internet gateway collects status information about existing flows, as well as new requests to get service.
- In algorithm 2, we consider the flows in an increasing order of class priority from class 1 to class K.
- Within each DiffServ class, the flows are considered for admission in some arbitrary order. Once a flow is admitted, the system uses the PDD scheduling algorithm to schedule the delivery of packets.
- The call admission scheme makes the decision by checking whether the new request can be admitted at a prescribed rate R (R is the x-axis in the figures), and $\frac{E_b}{I_0}$. If it is estimated that equation (2.5) holds when the new flow is served then the flow is admitted. Otherwise, the new mobile is rejected.

Algorithm 2 summarizes the main steps taken by the proposed CAC.

5.4 Simulation Results using the Basic CAC Algorithm

In this section, we present the simulation results assuming the use of Algorithm 2.

5.4.1 Average Delay

We illustrate the average delay results in figure 5.5 and figure 5.6. We observe that the results on average class delay confirm our intuition that using the CAC scheme decreases the average class delay by regulating the number of traffic flows in the system, reducing the interference, and keeping the congestion at relative low level.

Algorithm 2 Call Admission Control Algorithm With Single Transmission Rate

parameters:
K: the number of classes
N: the number of mobiles
f(t): ratio of the intra-cell to inter-cell interference

algorithm:
for (class j from 1 to K) **do**
  for (mobile i in class j waiting for admission) **do**
    Let N be the set of already admitted mobiles plus the new mobile;
    if ($\sum_{i=1}^{N} C_i < \frac{1}{1+f(t)}$ still hold) **then**
      mobile i is admitted
    else
      mobile i is rejected
    end if
  end for
end for

Both models do not experience congestion in the region from 16 Kbps to 64 Kbps as the system is at the light system load. In figure 5.6, there is still no heavy congestion beyond the light load region. The CAC scheme keeps the system out of heavy congestion so as to provide improved average delays. Lower priority classes (1 *and* 2) output better delay performance than that without CAC scheme although they experience heavier congestion than Class1. This consequence is also reflected from the throughput results. The average delay of Class1, for example, shows us that there are even no queue built up because Class1's average delay is always below 500 ms.

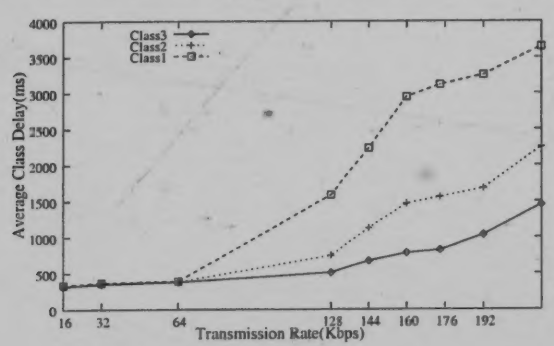

Figure 5.5: Average Delay without the CAC Algorithm

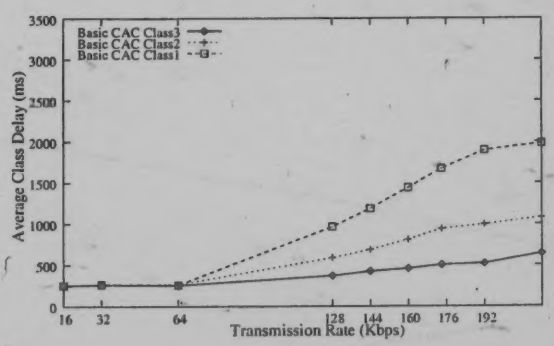

Figure 5.6: Average Delay with the Basic CAC Algorithm

5.4.2 Throughput

We present the throughput performance of the two models in figure 5.7, 5.8. Figure 5.7 shows the results without call admission control. Figure 5.8 shows the results with call admission control. We find that the throughput of Class3 and Class2 are always at the

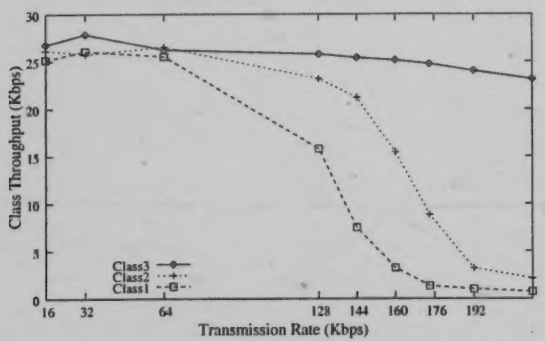

Figure 5.7: Throughput without the CAC

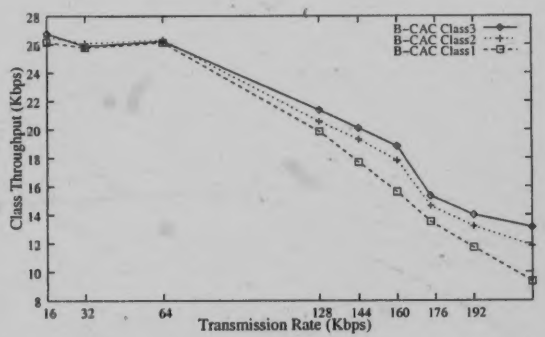

Figure 5.8: Throughput with the Basic CAC

same level, and the throughput of Class1 is at a lower level. However, it is clear that the performance of Class1 and Class2 with CAC is much better than that without CAC. It is because the CAC scheme brings fairness into system that weakens the greedy characteristic of the highest priority class, and let other classes have more service. Using CAC, each class can share part of entire system throughput, and dropping probability of packets in lower priority classes decreases.

5.5 Conclusions

In this chapter, we proposed a basic PDD scheduler and a basic CAC algorithm. We find that the use of CAC achieves a better approximation of the target class delay ratios, where the approximation is best when the traffic load is moderate (e.g. the interval [160 - 176]). This improvement comes at the expense of degradation of the throughput of the highest priority class.

Chapter 6

An Improved CAC Algorithm

In the previous chapter, we presented a simple CAC algorithm based on assigning the same rate to all mobiles. In this chapter, we seek to find an improved CAC algorithm intended to increase throughput. In section 6.1, we present the main ideas. Section 6.2 presents the simulation results.

6.1 Improved Call Admission Control Scheme

The algorithm presented in the previous chapter is based on a conservative estimation of the uplink capacity in terms of the maximum number of continuously transmitting mobiles the system can afford simultaneously. For example, if we assume the target $(\frac{E_b}{I_0})$ under perfect power control is 7 dB, and the transmission rate $R = 128$ Kbps, the maximum number of mobile users that the system can support is 6. Then, the total used uplink capacity is less than 770 Kbps. However, when the transmission rate $R = 32$ Kbps, the maximum number of mobile users that the system can afford is more than 20 (for a used capacity of 640 Kbps). Therefore, if we dynamically adjust the transmission rate according to the current traffic load, we may be able to admit more new requests and improve throughput. Algorithm 3 aims at using this idea.

The improved CAC scheme attempts to serve a new request by having some of the mobiles who have already been admitted reduce their transmission rate. The extent to which a mobile host reduces its transmission rate, as compared to other mobiles in the system, is

determined by the PDD parameters.

The main ideas can be explained as follows:

1. Suppose that the system has admitted 6 mobiles, and each one has transmission rate $R_{max} = 128$ Kbps. The system reaches its limit at this moment. When a new request is received, the system will conduct at most MAXITER iterations (e.g. MAXITER = 3). In each iteration, we randomly choose one admitted mobile that belongs to a certain class, for example Class3 (for which $\Delta_3 = 1$, for example). The chosen mobile backs off its transmission rate to $128 - 128 * \frac{1}{(2*1)^1} = 64$ Kbps. We re-check the interference constraints. If a feasible solution is found, the new request is admitted. Otherwise, the procedure repeats until all trials are made. A mobile can be chosen for rate reduction only once in the main loop.

2. Once the new request is admitted and starts transmitting data, it stays in a relative low transmission rate phase that we call warm-up phase. During this warm-up phase, it is possible that the system has a residual unused capacity, denoted C_{res} as indicated by the equation:

$$C_{res}(t) = \left(1 - (1 + f(t)) \sum_{i=1}^{N} \frac{1}{\frac{W}{R_{i,t} \cdot \zeta_i} + 1}\right) \geq 0 \qquad (6.1)$$

   where the ratio $f(t)$ of intra-cell interference to inter-cell interference is constant.

3. If the remaining capacity C_{res} is bigger than a minimum level ($C_{threshold}$), then C_{res} is assigned according to the PDD parameters.

4. If any mobile finishes its transmission, the system will have more capacity to assign; the most recent admitted mobile jumps to a new phase with higher transmission rate.

By adopting the two parts mentioned above together, the new CAC scheme is expected to increase the system throughput.

Algorithm 3 Call Admission Control Algorithm With Multi Rate

parameters:
K: the number of classes.
factor: loop control and rate deduction parameter.
MAXITER: 3.
R_{max}**: maximum rate that can be assigned to a mobile.**
$f(t)$**: ratio of intra-cell to inter-cell interference.**

algorithm:
for (mobile r making a new request) **do**
  Set $R_r = \frac{3}{4}R_{max}$. Add mobile r to the active set
  if $(\sum_{i\ is\ active} C_i > \frac{1}{1+f(t)})$ **then**
    factor = 0;
    while $(factor < MAXITER)$ **do**
      Choose one of the already admitted mobiles to reduce its rate, let i be the index of the chosen mobile, and let j be the delay class of mobile i.
      factor = factor + 1;
      $R_i(t) = R_i(t) - R_i(t) \cdot (2\Delta_j)^{(-factor)}$;
      if $(\sum_{i\ is\ active} C_i < \frac{1}{1+f(t)})$ **then**
        mobile i is admitted.
        break; {we have a feasible solution}
      end if
    end while
  end if
  if $(factor \geq MAXITER)$ **then**
    mobile r is removed from active set.
  end if
end for
if $(C_{res} \geq C_{threshold})$ **then**
  $C_{i,j} = C_{i,j} + \frac{\Delta_i}{\sum_{j=1}^{K} \Delta_j} \cdot C_{res}$
end if{Re-allocate residual resource according to delay control parameters ($\Delta \in (1, 2, 4)$,eg). Each mobile i in certain service class j will get some more capacity from C_{res}.}

6.2 Performance Comparison of Two CAC Schemes

In this section, we compare the performance of the new algorithm with the one presented in chapter 5. We first present the simulation results on average class delays in Figures 6.1, and 6.2.

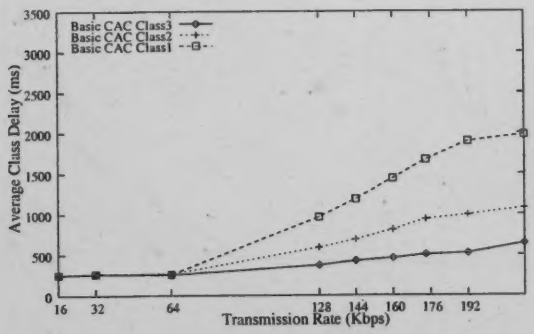

Figure 6.1: Average Delay of Algorithms 1 and 2 (same as Figure 5.6)

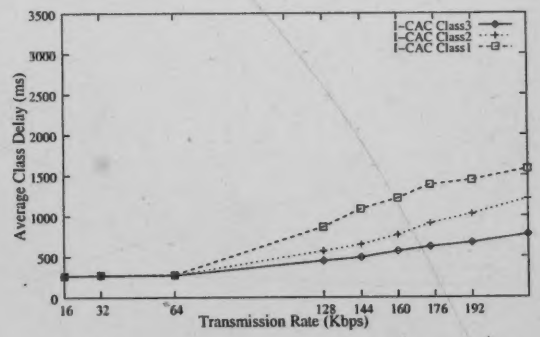

Figure 6.2: Average Delay of Algorithms 1 and 3

From the above figures, we find that the average class delays of both Class2 and Class3 are a bit higher than those with the basic CAC of algorithm 2, but, they are still in a region that is acceptable. With the improved CAC scheme, Class 2 and Class3 contribute more bandwidth than Class1 by backing off their transmission rates according to their proportional control parameters Δ_i in order to admit a new request. This dynamic transmission rate adjustment causes Class2 and Class3 to experience a bit higher average delay, and let Class1 keep as high transmission rate as it can, which leads to Class1's average delay at a low level.

6.2.1 Throughput

A remarkable feature that we found about the improved CAC scheme is its throughput performance. Figure 6.3 illustrates throughput results of the Basic CAC algorithm. Figure 6.4 shows the performance result from improved CAC.

Compared to the results in figure 6.4, we observe the improved CAC scheme brings even more fairness among classes. The throughput of high priority Class3 is not always at

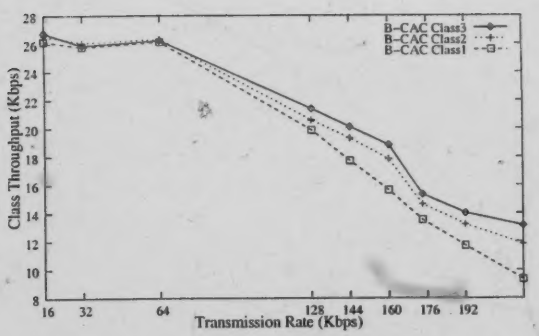

Figure 6.3: Throughput with the Basic CAC Algorithm

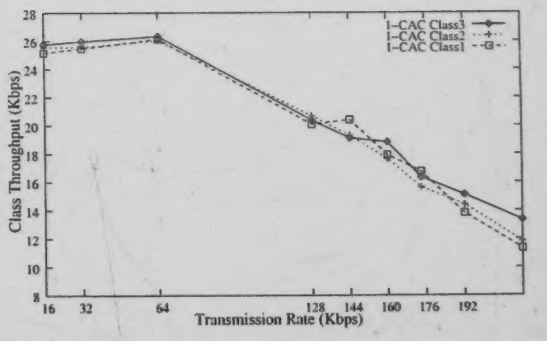

Figure 6.4: Throughput with the Improved CAC Algorithm

a high level any more. In certain circumstances, the throughput of Class1 is even higher than of Class3. Improved CAC algorithm gives a way to achieve fairness among all priority classes.

6.3 Conclusions

In the previous chapter we observe that the use of CAC results in improved average class delay and throughput. In this chapter we devise a CAC that aims at increasing throughput by accepting more requests while temporarily reducing the allocated rates of some already admitted mobiles. The simulation results obtained in this chapter show further improvements in average class delay and throughput where the lower priority classes benefit the most.

Chapter 7

Performance Enhancement Using Adaptive Parameter Setting

In the previous two chapters we developed and investigated the performance of a packet scheduling algorithm in conjunction with two different admission control algorithms. The obtained results show that the achieved proportional delay ratios fall below the required levels in situations where the system is not heavily loaded (e.g., the range [128 Kbps, 160 Kbps] of Figure 5.3). In such cases, the scheduler gives a good treatment to low priority classes. One may argue that under such traffic loads, it may be desirable for a network provider to exercise more control in allocating the available bandwidth so as to get better approximation of the desired proportional delays. This may be done by making more resources available to the high priority classes.

In this chapter we seek to achieve tighter control on delay ratios by focusing on the dynamics of the priority queueing taking place at the users side. In particular, we try to relate the relative weights used by the scheduling algorithm to induce service differentiation with the resulting relative delays. Unfortunately, we do not know of any queueing theoretic relation that works efficiently and can model the dynamics of the scheduling algorithm executing at the base station accurately. So, we develop our strategy using results from a simplified queueing model that captures the important factors.

To serve the above purpose, Section 7.1 identifies the *time-dependent priority* queues (*TDPQ*) as a suitable starting point for our design. An overview of the relevant results

on TDPQ are also given in section 7.1. Section 7.2 explains how we use the results, and discusses (in general terms) the suitability of the TDPQ model. Section 7.3 focuses on the computational aspects of the newly devised packet scheduler. Section 7.4 presents a few concrete numerical case studies to present the achieved performance gains. Section 7.5 presents results on the achieved average delay ratios, average delays, and throughput.

7.1 Time-Dependent Priority Queues (TDPQ)

Our design in this chapter uses analytical results on the average delays incurred by customers in a *time-dependent priority* queueing (*TDPQ*) system with N classes of customers. Roughly speaking, the result is used to approximate the achieved delay ratios obtained by our basic scheduling algorithm. To introduce the results on time-dependent priority scheduling system we follow [14] very closely.

The system is a single server non-preemptive queueing system with N classes of customers. The higher the index i, $i \in [1, N]$, of a class the higher the priority of the class. Customers in each class wait in a FCFS queue. Each queue is an M/M/1 queue. The relative priorities of the classes are determined by a user-defined set of control parameters $0 \leq b_1 \leq b_2 \leq \cdots \leq b_N$. The priority of a class i customer that has been waiting for ΔT sec is $\Delta T b_i$. When the server becomes available, the customer with the highest priority is chosen and served to completion. For class i let

- λ_i denotes the average arrival rate,
- $\overline{x_i}$ denotes the average service time,
- $\overline{x_i^2}$ denotes the second moment of the service time, and
- $\rho_i = \lambda_i \overline{x_i}$ (for $\rho_i < 1$) denotes the fraction of time the server is busy with customers from class i.

In addition, [14] defines the following quantities

- $\lambda = \sum_{i=1}^{N} \lambda_i$

- $\overline{x} = \sum_{i=1}^{N} \frac{\lambda_i}{\lambda} \overline{x_i}$, and
- $\rho = \lambda \overline{x} = \sum_{i=1}^{N} \rho_i$: the fraction of time the server is busy (for $\rho < 1$).

The following quantities and relations summarize the relevant results derived in [14].

1. Define W_0 to be the remaining service time (also called the residual life) of the customer found in service upon a new arrival's entry. Equation 3.7 (page 109) of [14] states that:

$$W_0 = \sum_{i=1}^{N} \frac{\lambda_i \overline{x_i^2}}{2}. \tag{7.1}$$

2. Define W_i, $i \in [1, N]$, to be the steady state average waiting time for class i customers. (The delay experienced by a class i customer is viewed as the sum of W_0, the delay due to other customers he finds upon his arrival, and the delay due to customers who arrive after he does.) Equation 3.48 (page 131) of [14] states that:

$$W_i = \frac{\frac{W_0}{1-\rho} - \sum_{k=1}^{i-1} \rho_k \cdot W_k \cdot [1 - (\frac{b_k}{b_i})]}{1 - \sum_{k=i+1}^{N} \rho_k \cdot [1 - (\frac{b_i}{b_k})]} \tag{7.2}$$

3. The *conservation law* for any M/G/1 queue and any non-preemptive work-conservative queueing discipline is (relations (3.16), page 114 of [14])

$$\sum_{i=1}^{N} \rho_i W_i = \begin{cases} \frac{\rho W_0}{1-\rho} & \rho < 1 \\ \infty & \rho \geq 1 \end{cases} \tag{7.3}$$

So, by fixing the control parameters $(b_i | \ i = 1, \cdots, N)$, one can achieve some desired average delays for each class. Our use of the above relations is the converse of the above: we want to compute suitable values of the b_is control parameters that yield a set of desired delay ratios.

7.2 Using the TDPQ System

We use the TDPQ system mentioned above in the following way, as follows.

- All mobile hosts that belong to the same DiffServ class k are assigned the same (unknown) weight value b_k.
- We note:
  - All classes have the same $\overline{x}_i$ value, denoted $\overline{x}$.
  - For simplicity, we assume $\overline{x}$ is normalized to one unit of time. Since the service time distribution is exponential, it follows that $\overline{x_i^2} = 2(\overline{x})^2 = 2$. Hence, $W_0 = \sum_{i=1}^{N} \frac{\lambda_i \overline{x_i^2}}{2} = \sum_{i=1}^{n} \lambda_i = \lambda$, and for each class $\rho_i = \lambda_i \overline{x} = \lambda_i$.

Discussion. In this part we address the suitability of using the TDPQ to find a suitable set of the control weights (the b_is) to achieve the desired average delay ratios by noting the following similarities and differences.

1. (a similarity) Both the TDPQ system and our basic packet scheduling algorithm compute the customer/packet priority by multiplying its current waiting time by a scalar, and serving the customer/packet with the highest product.
2. (a similarity) Assuming an equal number of mobile hosts in each DiffServ class, and a moderate overall traffic load (e.g., as in the range of [128 kbps, 160 kbps] of Figure 5.3), the basic scheduling algorithm gives a good treatment for low priority classes (and hence the achieved average delay ratios $(W_i/W_N | \ i = 1, 2, \cdots, N-1)$ falls below the required ratios. Under similar circumstances, the TDPQ system is also expected to serve the low priority classes well.
3. (a similarity) Assuming an equal number of mobile hosts in each DiffServ class (as above), and a heavy overall traffic load (e.g., as in the range [176kbps, 192kbps] of Figure 5.3), the basic scheduling algorithm discriminates against the low priority classes, and many packets in such low priority classes are ignored (dropped out before transmission). Under similar circumstances, the TDPQ system is also expected to discriminate against the low priority classes.

4. (a difference) The TDPQ system is a single server, however, the base station in the basic scheduling algorithm provides multiple servers.

We conclude that the TDPQ provides a reasonable way for use in estimating the values of the control parameters that derive the basic scheduling algorithm.

7.3 Solving For the Required Delay Ratios

This section outlines the details of the computations needed to solve the system to find suitable b_i. We use N to denote the number of DiffServ classes (rather than the mobiles as in the previous chapters). Given the required $N-1$ delay ratios $(\frac{\Delta_i}{\Delta_N} | i = 1, 2, \cdots, N-1)$ we would like to solve the above system (equations (7.2)) of nonlinear equations to find suitable values for the control parameters $(b_i \mid i = 1, 2, \cdots, N)$ to achieve the desired proportional delays (i.e., $\frac{W_i}{W_N} = \frac{\Delta_i}{\Delta_N}$, for $i \in [1, N-1]$). The solution involves the following basic steps

1. We write equation (7.2) as follows:

$$W_i - W_i \sum_{k=i+1}^{N} \rho_k + W_i \sum_{k=i+1}^{N} \rho_k (\frac{b_i}{b_k}) = \frac{W_0}{1-\rho} - \sum_{k=1}^{i-1} \rho_k W_k + \sum_{k=1}^{i-1} \rho_k W_k (\frac{b_k}{b_i}),$$

and define:

- $A_i = W_i \cdot \sum_{k=i+1}^{N} \frac{\rho_k}{b_k}$

  (the third term on the *L.H.S* after factoring b_i out), note that for $i = N$ the summation equals zero and $A_N = 0$;

- $B_i = \sum_{k=1}^{i-1} \rho_k \cdot W_k \cdot b_k$

  (the third term on the *R.H.S* after factoring $\frac{1}{b_i}$ out);

- $C_i = \frac{W_0}{1-\rho} - \sum_{k=1}^{i-1} \rho_k \cdot W_k - (1 - \sum_{k=i+1}^{N} \rho_k) \cdot W_i$

  (the remaining terms after moving terms 1 and 2 from the L.H.S to the R.H.S).

2. So, condition (7.2) can be rewritten as

$$A_i \cdot b_i - B_i / b_i - C_i = 0. \tag{7.4}$$

3. Define $\delta_i = \frac{W_i}{W_N}$ for $i \in [1, N]$. We now observe that the conservation law (7.3) ($\sum_{i=1}^{N} \rho_i W_i = \frac{\rho W_0}{1-\rho}$) allows us to express each W_i in terms of the δ_i. To see this, we first write the conservation law as: $W_N(\sum_{i=1}^{N} \rho_i \frac{W_i}{W_N}) = \frac{\rho W_0}{1-\rho}$, which yields:

$$W_N = \frac{\frac{\rho W_0}{1-\rho}}{\sum_{i=1}^{N} \rho_i \delta_i}.$$

Since $W_i = \delta_i W_N$ (by definition of δ_i), it then follows that:

$$W_i = \frac{\delta_i \cdot \frac{\rho \cdot W_0}{1-\rho}}{\sum_{i=1}^{N} (\rho_i \cdot \delta_i)}. \tag{7.5}$$

4. The following steps apply the *successive over relaxation* (SOR) [9] method to solve the system of nonlinear equations iteratively, until a convergence condition is satisfied. The method uses a chosen constant $\omega, \omega \in [1, 2]$. As shown in algorithm 4, each iteration computes new values for the vector $\mathbf{b} = (b_1, b_2, ..., b_N)$.

5. In particular, each iteration computes

$$b_i^{(k+1)} = (1-\omega)\, b_i^{(k)} + \omega\, \varphi_i(b_1^{(k+1)}, b_2^{(k+1)}, ..., b_{i-1}^{(k+1)}, b_i^{(k)}, b_{i+1}^{(k)}, ..., b_N^{(k)}) \tag{7.6}$$

for $i = 1, 2, ..., N$, where the function $\varphi_i(\mathbf{b})$ is defined as follows:

- For $1 \leq i < N$, from equation (7.4), we have

$$\varphi_i(\mathbf{b}) = \frac{C_i + B_i/b_i}{A_i},$$

- For $i = N$, $A_N = 0$. Equation (7.4) implies that $b_N = -\frac{B_N}{C_N}$. We rewrite it by factoring a b_N on both sides. It gives us: $b_N^2 = -\frac{b_N B_N}{C_N}$. Since b_N must be positive, we get

$$\varphi_N(\mathbf{b}) = \sqrt[2]{\frac{B_N b_N}{(-1) \cdot C_N}}$$

6. Defining the SOR convergence condition as:

$$\| \sum_{i=1}^{N} f_i(\mathbf{b}^{(k+1)}) - \sum_{i=1}^{N} f_i(\mathbf{b}^{(k)}) \| < \epsilon, \tag{7.7}$$

where, $f_i(\mathbf{b}) = A_i b_i - B_i/b_i - C_i$, and ϵ is a pre-defined error threshold (e.g. $\epsilon = 0.0002$).

7. Algorithm 4 shown below is based on the above analysis.

Algorithm 4 SOR Iterative Method to Find Right Control Parameters Δ

parameters:
ω**: SOR convergence factor**
ρ**: System load,** $\rho = \sum_{i=1}^{N} \rho_i$

algorithm:
$k = 0, b_i^{(0)} = \frac{1}{W_i} = 1/(\frac{\delta_i \cdot \frac{\rho \cdot W_0}{1-\rho}}{\sum_{i=1}^{N}(\rho_i \cdot \delta_i)})$; # *a guess for* b_i
while $(\| \sum_{i=1}^{N} f_i(b^{(k+1)}) - \sum_{i=1}^{N} f_i(b^{(k)}) \| > \epsilon \;\&\&\; k < MAX_ITERATION)$ **do**
  for $i = 1; i \leq N; i++$ **do**
    $b_i^{(k+1)} = (1-\omega)\, b_i^{(k)} + \omega\, \varphi_i(b_1^{(k+1)}, b_2^{(k+1)}, ..., b_{i-1}^{(k+1)}, b_i^{(k)}, b_{i+1}^{(k)}, ..., b_N^{(k)})$;
  end for
  $k = k + 1$; # *update iteration counter*
end while

7.4 Case Studies

In this section we present three numerical case studies that illustrate the possible improvements obtained from using the new adaptive scheduling algorithm, as well as the convergence rate of the iterative solution method.

Case 1.

Here, we would like to achieve target delay ratios of $\Delta_1 : \Delta_2 : \Delta_3 = 4 : 2 : 1$, where each mobile user is assigned a rate of 144kbps. We use three DiffServ classes, and set $\rho_1 = 0.3$, $\rho_2 = 0.25$, and $\rho_3 = 0.25$ ($\rho = 0.8$). To generate traffic that makes the system work at the desired value $\rho = 0.8$, we calculate the maximum number of mobile users that the system can support, $N_{max} = 6$, and let the number of mobiles for each class be $N_1 = 4$, $N_2 = 3$, and $N_3 = 3$. By solving the above non-linear system of equations, we get $b_1 : b_2 : b_3 = 10.18 : 2.95 : 1$ in 10 iterations. Applying the new control parameters to the scheduling algorithm, we get the improved results illustrated in Figure 7.2. We observe that target delay ratios are achievable even when the system is under moderate load.

Case 2.

In order to verify the performance of applying suitable control parameters b_i, we con-

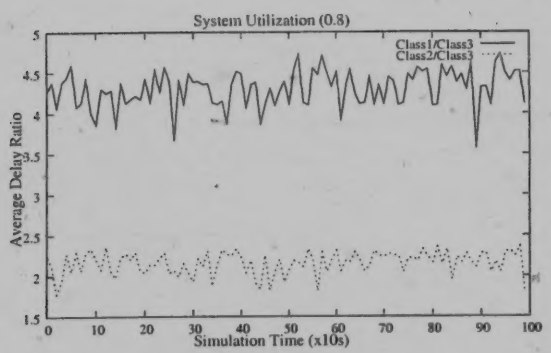

Figure 7.1: Results using the Δ_i values.

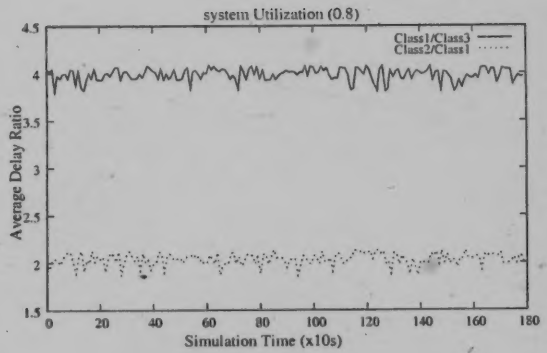

Figure 7.2: Results using the b_i values

sider a second case study. At this time, we intend to achieve target delay ratios of $\Delta_1 : \Delta_2 : \Delta_3 = 16 : 4 : 1$, where each mobile user is assigned a rate of 160 kbps. For the classes, we want to set $\rho_1 = 0.35$, $\rho_2 = 0.3$, and $\rho_3 = 0.3$ ($\rho = 0.95$). To generate traffic that give this ρ, we compute the maximum number of mobile users $N_{max} = 5$, and let the number of mobiles for each class be $N_1 = 4$, $N_2 = 3$, and $N_3 = 3$. Solving for the b_i's, we get $b_1 : b_2 : b_3 = 35.93 : 5.11 : 1$ in 13 iterations. Applying the new control parameters to the scheduling algorithm, we observe improved results as illustrated in Figure 7.4. We observe that wider target delay ratios can be attained when the system is running at heavy load.

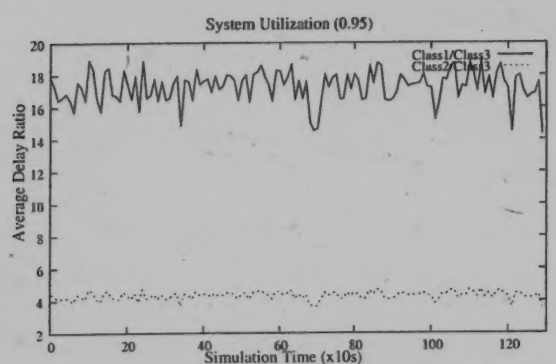

Figure 7.3: Results using the Δ_i values.

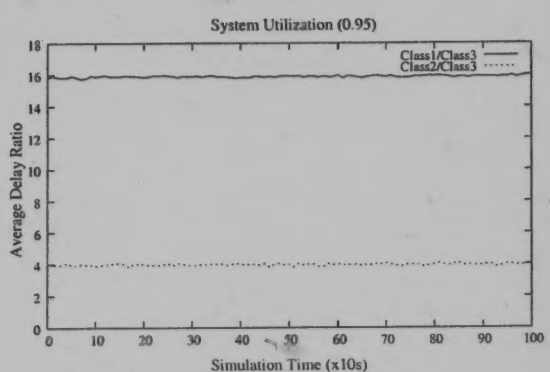

Figure 7.4: Results using the b_i values

Case 3.

In this case we have: $\rho_1 = 0.2$, $\rho_2 = 0.2$, $\rho_3 = 0.2$ ($\rho = 0.6$), each mobile is assigned

a rate of 144 kbps, the target delay ratios are $\Delta_1 : \Delta_2 : \Delta_3 = 4 : 2 : 1$, and the number of mobiles of class 1, 2, and 3 are $N_1 = 3$, $N_2 = 3$, and $N_3 = 4$, respectively. In this case, our SOR algorithm does not converge to a solution. The reason is when the system is under light load condition, all the class members can get service.

7.5 Results on Average Delay Ratio, Average Delay and Throughput

The previous section illustrates the performance of the adaptive scheduling algorithm. In this section, we compare the performance of the new algorithm versus the performance of algorithm 1.

In all the figures shown below, the x-axis reflects the traffic load. The system runs at light system load when mobiles transmit in the range [16 kbps, 64 kbps]. No congestion is expected in this range. From 128 kbps to 196 kbps, the system runs at heavy load. Congestion occurs due to packets backlogged in each class. Above the transmission rate 196 kbps, the system goes into extremely heavy load . In this range, more packets go into the system and be backlogged in the queues, but less packets can be served due to resource competition. Most of the packets are finally dropped from queues. Therefore, the average delay of each class is extremely high and no guarantee of proportional delay can be made.

7.5.1 Average Delay Ratios

Figure 7.5 illustrates both the performance of algorithm 1 and algorithm 4 in terms of average delay ratio. We note that algorithm 4 outperforms algorithm 1 in achieving more accurate delay ratios. Even when system utilization is relative moderate, the achieved delay ratio is around 2.01 of Class2 over Class3, and 3.99 of Class1 over Class3. However, when the system goes toward heavier load conditions (i.e. in the range [128 kbps, 196 kbps]), unimproved PDD method can also reach better proportional delay ratio, which are around 3.98 and 1.99, respectively for $Class2/Class3$ and $Class1/Class3$. This performance results reflect what we expect and describe in previous section.

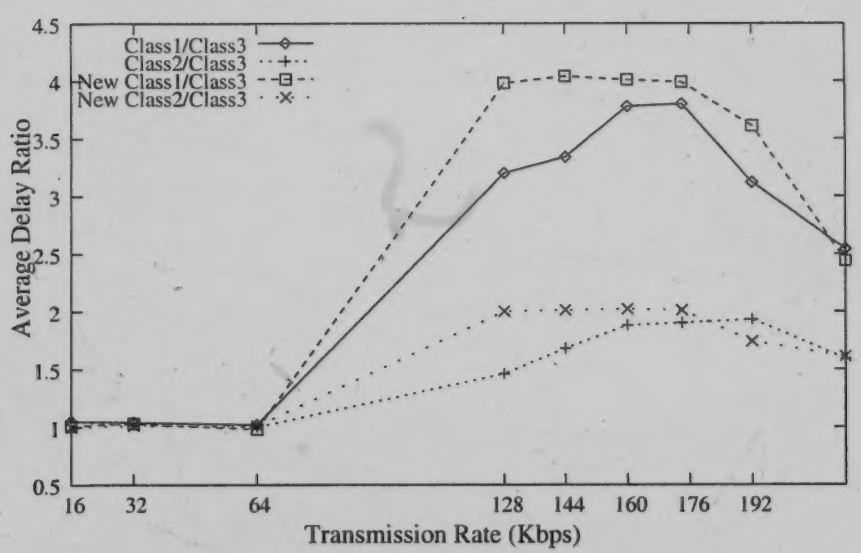

Figure 7.5: Average Delay Ratio under Different System Load

7.5.2 Average Delay of Classes

Figure 7.6 shows the achieved average delay of each class. We observe that, although algorithm 4 achieves target delay ratios by using $b_i's$ as control parameters, it increases the average delay of each class. At 128 kbps, for example, the average delay of Class1 from algorithm 4 is almost 1100 ms higher than that of Class1 from algorithm 1.

The improved PDD method causes increased delay increase because it extends the delay span of each of packets that belong to lower priority classes (Class2, Class1) by using new adjusted control parameters (eg. $1 : 2.95 : 10.18$ instead of $1 : 2 : 4$). Consequently, more packets will be backlogged in lower priority classes (Class2, and Class3) than higher priority classes (Class1). Only can a small part of these backlogged packets be served. Furthermore, most of these served packets have to wait longer than packets do in unimproved PDD method, and some almost reach their expiration limit when they get service. A big part of the packets backlogged in queues can not get service in time, and eventually be dropped. Consequently, it causes a worse throughput performance.

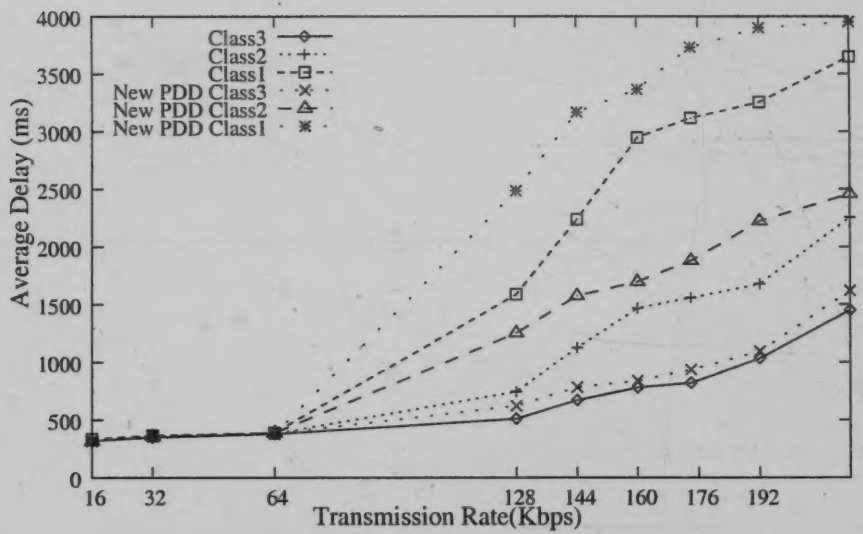

Figure 7.6: Average Delay under Different System Load

7.5.3 Throughput of Classes

In figure 7.7 and 7.8, we find that the three classes show no notable difference of their throughput when the system is running at light system load (16 kbps to 64 kbps) because each of three class share $\frac{1}{3}\rho$ as we described in section 4.2.3.

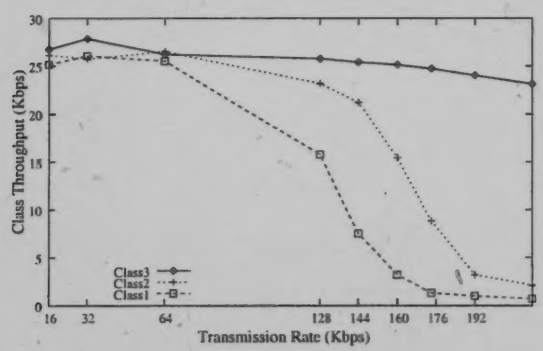

Figure 7.7: Class Throughput – Δ_i

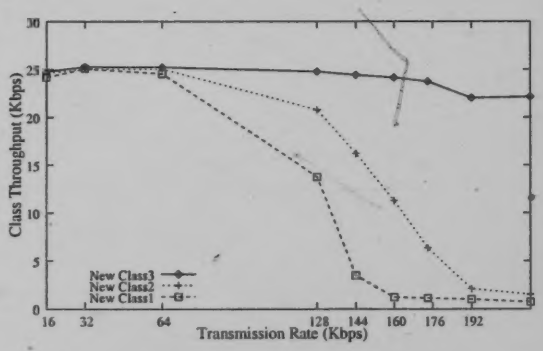

Figure 7.8: Class Throughput – b_i

Apparently, we can find that higher priority $Class3$ always gets service and maintains the throughput at the same level in both methods because of its high priority. Class1 and Class2 are hardly served when system goes toward heavy load condition as the mobiles

are served with higher transmission rates. The reason of this has been discussed in section 5.2.3.

However, we also find that the basic PDD model performs better than the improved PDD model on Class1 and Class2. Without admission control, all admitted traffic compete for the limited system resources. In order to maintain the lower delay in higher priority Class1 and precise target delay ratio, scheduling makes throughput of lower priority classes suffer. When system is running at heavy system-load situation, unfairness caused by congestion among traffic classes is notable.

7.6 Conclusions

We present an algorithm that solves a non-linear system of equations in order to find a set of delay control parameters that are likely to achieve the desired delay ratios. After applying the set of control parameters found by algorithm 4, the target average delay ratios have been shown to be achievable even when the system is under moderate traffic load. However, the shortcoming of adjusting control parameters in order to achieve precise target delay ratios is an increase in the average class delay.

Chapter 8

Conclusions and Future Works

8.1 Summary

Next generation wireless networks aim at enabling the end user to run many of the Internet multimedia applications. Achieving this task requires sharing of the available capacity among multiple users according to the needed QoS requirements in a fair and efficient manner.

In this thesis, we consider extending the DiffServ architecture to the uplink traffic of a 3G CDMA-based wireless environment. In particular, the thesis presents a framework to support the class-based proportional delay differentiation architecture. The proposed framework includes both scheduling algorithms and admission control algorithms. Using simulation, and assuming that traffic requests occur with equal probability among three DiffServ classes, we have shown the following main results.

- A basic scheduling algorithm (Algorithm 1) is capable of achieving accurate delay ratios when the system is moderately loaded.
- Incorporating a basic admission control algorithm (Algorithm 2) improves the performance further.
- Using a simple heuristic algorithm (Algorithm 3) to lower the rates of some mobiles in order to accommodate more newly incoming traffic results in further improvements.

- Using the TDPQ model to compute suitable settings for the scheduler control parameters extends the range of traffic loads under which accurate delay ratios can be achieved.

8.2 Future Works

In this section we discuss some possible research directions that complement the work done in the thesis.

1. We first note that the work done here concerns the case of equal traffic load in each DiffServ class. If the traffic of one class behaves in a greedy way then it can affect the performance of the system. This aspect calls for allocating resources so as to guarantee a minimum amount for each DiffServ class. More research is required to ensure dynamic sharing of resources while satisfying the above aspect.

2. The scheduling algorithms devised in the thesis aim at achieving proportional delay differentiation by allocating the same data rate R to each user. Algorithm 2 tries to lower the rate R temporarily for some users to better utilize the system resources. The obtained results show some improvements. It is therefore interesting to examine more sophisticated ways of achieving the desired delay differentiation by adaptively changing the assigned rates.

3. In chapter 2 we noted that distributed algorithms can be used to achieve power control. The use of such algorithms reduces the complexity of the base station at the expense of losing bandwidth (since the algorithms require time to converge to the minimum acceptable transmission power levels). Since the resulting systems are expected to be cheaper than the sophisticated UTRA networks it may be of interest to investigate the amount of degradation in system performance for the DiffServ delay classes under such circumstances.

4. The call admission control algorithms devised in the thesis admit new traffic requests based on the available resources at the instant of the decision making process. The

amount of available resources at future instants depends on the mobility pattern of the users. The reported results of the thesis are obtained assuming a random mobility model. In cases when mobility information about a subset of users are available, it would be desirable to incorporate such information in the decision making process. Some strategies to exploit such knowledge appear in the literature (specially for predicting handoffs). More work is needed to exploit such knowledge to provision QoS measures in a CDMA-based cellular environment.

Bibliography

[1] IETF group, an assured forwarding PHB. In *RFC 2597*, June 1999.

[2] IETF group, an expedited forwarding PHB. In *RFC 2598*, June 1999.

[3] 3GPP group, QoS concept and architecture. In *TS 23. 107, release 5*, January. 2002.

[4] IETF group, DiffServ and IntServ. In *http://www.ietf.org*, March 2003.

[5] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated services. In *RFC 2475 IETF*, December. 1998.

[6] H. J. Chao and X. Guo. *Quality of Service Control in High-Speed Networks*. John Wiley and Sons, New York, 2002.

[7] E. H. Dinan and B. Jabbari. Spreading codes for direct sequence CDMA and wideband CDMA cellular networks. *IEEE Communications Magazine*, pages 48–54, September 1998.

[8] C. Dovrolis, D. Stiliadis, and P. Ramanathan. Proportional differentiated services: Delay differentiation and packet scheduling. In *Proceedings of ACM*, pages 26–34, September 1999.

[9] G. H. Golub and J. M. Ortega. *Scientic computing and differential equations - An introduction to numerical methods*. Cambridge University Press, 1992.

[10] D. Goodman. *Wireless Personal Communications Systems*. Addison Wesley, Reading, Massachusetts, 1997.

[11] A. Jamalipour. *The Wireless Mobile Internet Architectures, Protocols, and Services*. John Wiley and Sons, West Sussex, England, 2003.

[12] L. Jorguseski, J. Farserotu, and R. Prasad. Radio resource allocation in third-generation mobile communication systems. *IEEE Communications Magazine*, pages 117–123, 2001.

[13] S. Keshav. *An Engineering Approach to Computer Networking: ATM Networks, the Internet and the Telephone Network*. Addison Wesley, Reading, Massachusetts, 1997.

[14] L. Kleinrock. *Queueing Systems Volume II: Computer Applications*. Johan Wiley & Sons, New York, 1976.

[15] Y.-C. Lai and W.-H. Li. A novel scheduler for proportional delay differentiation by considering packet transmission time. *IEEE Communications Letters*, 7(4):527–538, April 2003.

[16] C. Li, S. Tsao, M. Chen, Y. Sun, and Y. Yuang. Proportional delay differentiation service based on weighted fair queueing. In *Proceeding IEEE International Conference on Computer Communications and Networks*, pages 418–423, October 2000.

[17] S. Lu, V. Bharghavan, and R. Srikant. Fair scheduling in wireless packet networks. *IEEE/ACM Transactions on Networking*, 7(4):473–489, August 1999.

[18] T. Nandagopal, N. Venkitaraman, R. Sivakumar, and V. Bharghavan. Delay differentiation and adaptation in core stateless networks. In *INFOCOM*, pages 421–430, March 2000.

[19] K. Nichols, S. Blake, F. Baker, and D. Black. Definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers. In *RFC 2474 IETF*, December 1998.

[20] K. Nichols, V. Jacobson, and L. Zhang. A two-bit differentiated services architecture for the internet. In *RFC 2638 IETF.*, July 1999.

[21] A. K. Parekh and R. G. Gallager. A generalized processor sharing approach to flow control in integrated services networks: The single-node case. *IEEE/ACM Transactions on Networking*, 1(3):344–357, June 1993.

[22] T. Rappaport. *Wireless Communications Principles and Practice*. Prentice Hall, 2002.

[23] J. Schiller. *Mobile Communications*. Addison-Wesley, London, England, 2000.

[24] D. Shen. *Resource Management in Wireless CDMA Networks*. PhD thesis, Rensselaer Polytechnic Institute, Electrical, Computer and System Engineering, Troy, New York, October 2001.

[25] D. Shen and C. Ji. Admission control of multimedia traffic for third generation CDMA network. In *INFCOM 2000*, volume 3, pages 1077–1086, March 2000.

[26] D. Shen and C. Ji. CDMA network capacity over shadowing channels. In *12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, volume 1, pages 149–153, October 2001.

[27] D. Shen and C. Ji. Outage evaluation under traffic heterogeneity in CDMA networks. In *IEEE GLOBECOM '01*, volume 6, pages 3709 –3713, November 2001.

[28] D. Shen and C. Ji. Reverse link admission control for integrated services in CDMA networks. In *Wireless 2001, Calgary, Canada*, July 2001.

[29] W. Stallings. *Wireless Communications and Networks*. Prentice-Hall, 2002.

[30] L. Wu. *Delay Differentiation for Wireless Networks*. M.Sc. thesis, University of Alberta, Department of Computing Science, Edmonton, Alberta, Canada, August 2002.

[31] L. Xu, X. Shen, and J. W. Mark. Aggressive code-division generalized processor sharing for Qos guarantee in multimedia CDMA cellular network. In *IEEE GLOBECOM 2001*, volume 1, pages 951–955, November 2002.

[32] L. Xu, X. Shen, and J. W. Mark. Dynamic bandwidth allocation with fair scheduling for WCDMA systems. *IEEE Wireless Communications*, pages 26–32, April 2002.

[33] R. D. Yates. A framework for uplink power control in cellular radio systems. In *IEEE Journal On Selected Areas In Communications.*, volume 13 No. 7, pages 1341–1347, September 1995.

[34] H. Zhang. Service disciplines for guaranteed performance service in packet-switching networks. *Proceedings of the IEEE*, 83(10):1374–1396, October 1995.

[35] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala. RSVP: A new resource reservation protocol. *IEEE Network Magazine*, pages 8–18, September 1993.